MODELING THE DYNAMICS OF CARBON SEQUESTRATION: INJECTION STRATEGIES, REMOBILIZATION, AND CAVITATION

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

In Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Erik James Huber

December 2017

MODELING THE DYNAMICS OF CARBON SEQUESTRATION: INJECTION STRATEGIES, REMOBILIZATION, AND CAVITATION

Erik James Huber, Ph.D.

Cornell University 2017

Carbon sequestration involves capturing CO2 from an exhaust source, compressing it to a supercritical fluid state, and injecting it underground where it can be stored. The environmental engineering goal of carbon sequestration is to prevent further increases in atmospheric CO2 concentration levels. This dissertation examines three aspects of carbon sequestration pertaining to the fluid dynamics of injection and post-injection within the geologic subsurface.

First, a time dependent injection strategy of brine alternating with CO2 is proposed as a method to reduce the CO2 mobility by increasing the rates of residual trapping and dissolution. After making some assumptions, the equations governing the dynamics of CO2 mass transport become a coupled set of 1D wave equations, whose wave speeds provide insight into the relative permeability conditions required for this injection strategy to be most effective. Numerical solutions using the method of characteristics are then compared against 3D TOUGH2 simulations and comparable favorably to one another. Both models predict that alternating brine injection can reduce the mass fraction of mobile CO2 to less than 10% using a volume ratio brine:CO2 of less than 2.75 and on time scales that are $10^2 – 10^4$ times faster than would occur with a continuous injection of CO2.

Second, the stability of residually trapped CO2 is analyzed with respect to its susceptibility to become remobilized. Here, a reservoir containing a region of

homogeneously dispersed pockets of residually trapped CO2 is considered. Should the pore pressure decrease, the CO2 will expand, remobilize, rise within the domain, and potentially spread along the caprock of the formation. The dynamics of this process are predicted using two different relative permeability models: Brooks-Corey, and a modified Brooks-Corey that incorporates percolation theory. Experimental data justifying this latter model is presented. More importantly, the time scales of remobilized CO2 motion are shown to be vastly different for these two models and suggest the need for further experimental data.

Finally, as dry CO2 is injected into brine occupied reservoirs, there exists the potential for mechanical tension forces to be generated within the brine which can cause cavitation to occur. These cavitation dynamics are modeled here using explicit and averaged equations and compared to experimental data from drying an idealized synthetic heterogeneous porous media.

BIOGRAPHICAL SKETCH

Erik James Huber was raised in Fulton, NY by his parents, Cindy and Jim. Throughout grade school both he and his younger brother, Mark, excelled at intellectual endeavors, casual scientific exploration, and driving their teachers mad with ceaseless inquisition. Erik graduated 3rd in his class from G. Ray Bodley High School in 2004 and attended Alfred State College on a full scholarship. He graduated from Alfred State in 2006 with an associate's degree in Engineering Science and continued his undergraduate studies at the State University of New York at Buffalo, where he earned his bachelor's degree in Mechanical Engineering in 2008. Declining a full time engineering position at his place of internship, Erik decided instead to pursue graduate school. He first attended Columbia University and earned his M.S. in Mechanical Engineering in 2010. Deciding to end his stint in the big city, he was admitted to Cornell University that same year, where he began his doctoral studies under the direction of Abe Stroock and Don Koch. Erik defended his Ph.D. from Cornell University in 2016 and at the time of this writing plans to start his career in industry and hopes to one day transition back to academia to complement his lifelong passion for learning.

To my family for their endless support,

and to my amazing wife, whom I blame for making my doctorate the second most fulfilling part of my time here at Cornell.

14	11	19	85
86	18	8	17
9	16	87	17
20	84	15	10

129

ACKNOWLEDGMENTS

Not only does this work embody the efforts of many years of study, it also is the product of input and guidance from countless individuals. It should come as no surprise that I first acknowledge my co-advisers, Abe Stroock and Don Koch. Each of these gentlemen are true experts in their craft and I am forever grateful for their insight, patience, and thoughtfulness that they offered me throughout my Ph.D. It has been a pleasure and an honor to work with them so closely over all these years and to try and absorb as much from them as I could. None of my work would have been possible without their support.

I would also like to thank so many members of the Cornell faculty with whom I have interacted during my studies. In particular, I thank Jeff Tester, Julie Nucci, Paulette Clancy, Terry Jordan, and Olivier Desjardin. These individuals gave me crucial opportunities to showcase my research, engage the scientific community, and experience the world in a way that I would have never been able to do on my own.

Last but not least, I thank my colleagues and peers. The people with whom I would have a beer (or two) and discuss research, the life of a grad student, and the day's recent events. Although these moments of friendship and encouragement will never make themselves present within the lines of this dissertation, I can assure you that this dissertation would not exist without them.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 An Introduction to flow in porous media

In the mechanical engineering discipline, undergraduate fluid mechanics generally focuses on scaling analysis; fluid statics; laminar, low Reynolds number flow (Stokes Equations); inviscid flow along streamlines (Bernoulli's Equation); and turbulent, high Reynolds number flow in pipes. While no single course can achieve complete inclusivity, one broad area that this traditional curriculum ignores is the analysis of flow in porous media. As a result, mechanical engineers pursuing graduate work in the field of flow through porous media must first begin at the introductory level to understand the general concepts, common nomenclature, and basic physics of flow in porous media before being able to contribute to novel research. The purpose of this chapter is to briefly summarize the most important points of this introductory knowledge base such that subsequent chapters can focus on the research of this dissertation. These concepts and topics are discussed in greater detail in (Bejan, 2006; Berkowitz and Ewing, 1998; Lage, 1998).

One obvious place to begin is to define the characteristics of porous media. A porous medium is comprised of an interconnected solid matrix whose structure does not occupy the entire interior volume. Instead, some fraction of the internal volume of the structure is empty space, called voids or void space. An example of a porous medium is presented in Figure 1d. The total fraction of void space contained within the porous medium is called porosity, ϕ [-], defined formally as

$$\phi = \frac{V_{pore}}{V_{tot}}, \qquad \textbf{Eq. (1)}$$

where V_{tot} [m^3] is the total volume enclosed by the outer most boundaries of the solid matrix and V_{pore} [m^3] is the total volume of empty space inside the domain defined by V_{tot}, and is the most basic property of a porous medium.

Countless materials are classified as porous media, such as soils, sands, foams, sponges, and wood. Because void spaces can exist throughout a material on submillimeter length scales ($\mathrm{nm} - \mu\mathrm{m}$) even many seemingly solid materials are technically porous, including rocks, ceramics, and concrete. Often times, these mineral-based materials are formed by the packing of irregular grains, which leave empty spaces in between grain boundaries. Some biological materials can also be modeled as porous material, such as the xylem in trees, which act as conduits for the transport of water from the roots to the canopy. In general, porous media exhibit two key features: (1) many continuous, interconnected pathways of void space exist within the domain to allow fluids to pass through the material and (2) the characteristic pore throat size is small relative to the characteristic size of the domain. As a result of these characteristics, it is generally difficult or impractical to use the Navier-Stokes equations to explicitly solve for the flow field everywhere in the domain because it requires perfect knowledge of the entire interior pore space (often impossible to obtain in geologic problems) and extremely large computational meshes to resolve the pore scale details of $\mathcal{O}(\mu\mathrm{m})$ throughout the global domain, which can often be $\mathcal{O}(\mathrm{km})$.

In order to analyze and predict the flow through porous media, some of the early models start with a base geometry, such as a tube or ball-and-stick (Berkowitz and

Ewing, 1998). This base geometry is then repeated in parallel or interconnected to form a global network of fluid conduits, shown in Figure 1a-c. Because the base geometry is well defined, some simple analyses can be performed to estimate the global flow properties. These well-defined geometries also make it easier to model the porous medium computationally. However, these models are often restricted in their domain size due to computational limitations and generally do not exhibit the types of geometric heterogeneity and complexity that exists in natural porous media. As a result, many of the properties used to describe porous media are fit to experimental data using models that treat the porous medium as a bulk domain, which ignores the details of the flow locally within the pores. It is in this spirit of macroscale analysis that the governing equations are derived and the method first used by Darcy to characterize the flow through porous media (Lage, 1998).

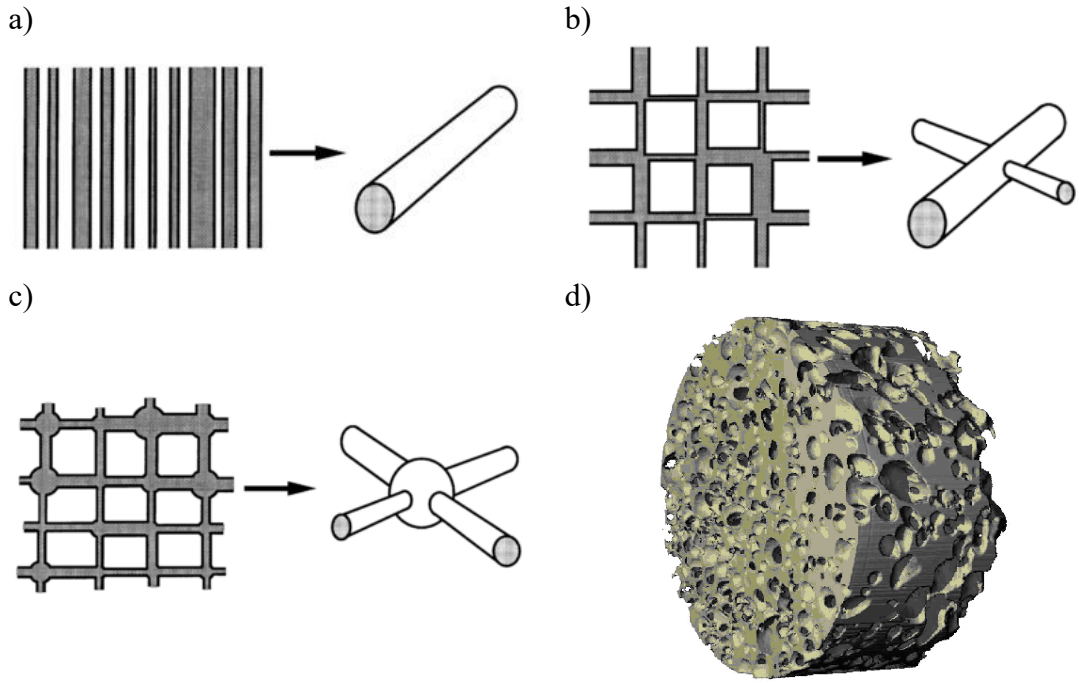


Figure 1. Three examples of the evolution of pore space models (Berkowitz and Ewing, 1998) (a) parallel tubes, (b) tube network, (c) ball-and-stick network. (d) A 3D scanning of a natural porous ceramic (Fey, 2015).

1.2 Basic equations

One of the first experiments conducted by Darcy on porous media was that of water flowing horizontally (perpendicular to the gravity field) through a sandy channel at steady state as a result of a global pressure gradient applied to the fluid. It was recognized that the volumetric flow rate, Q [$m^3 \cdot s^{-1}$], entering and leaving the sample was proportional to several properties, namely

$$Q \propto \frac{A}{L} \Delta P \qquad \textbf{Eq. (2)}$$

where A was the cross sectional area of the porous medium [m^2], L was the length of

the porous medium [m], and ΔP was the global pressure difference upstream and downstream of the porous sample [Pa] (Darcy, 1856; Freeze, 1994; Lage, 1998). To make equation 2 an equality, a proportionality constant was introduced, such that

$$Q = K\frac{A}{L}\Delta P. \qquad \textbf{Eq. (3)}$$

In this equation, K is the total permeability of the porous medium [$m^2 \cdot Pa^{-1} \cdot s^{-1}$] and functionally represents how easily a fluid can pass through the medium of unit length and unit cross-section. Originally, K was assumed to be a constant for a given type of material (Lage, 1998). However, upon repeating the experiments with fluids other than water it became obvious that equation 3 should be modified to

$$Q = \frac{\kappa}{\mu}\frac{A}{L}\Delta P \qquad \textbf{Eq. (4)}$$

where κ is the intrinsic permeability, or simply, permeability [m^2] and μ is the dynamic viscosity of the fluid [Pa·s]. By making this modification, the permeability is found to be independent of the choice of fluid and remains constant for a particular porous medium sample, which is to say that κ encodes purely geometric information about the solid host. For this reason, when modeling porous media as a network of parallel tubes of radius r_p, then κ is found to be proportional to r_p^2 ($\kappa \propto r_p^2$). Kappa also retains the same qualitative nature as K: to provide a measure of how readily fluid can pass through the medium per unit length and area. Materials with large permeability offer little flow resistance, while materials with small permeability offer high resistance to flow. After porosity, permeability is the second most important property of a porous medium. Because the flow resistance comes from the shear stress at the walls, porous media with

larger porosities tend to also exhibit larger permeability. Although this relationship is generally true and makes physical sense, no broad, fundamental correlation exists between porosity and permeability for natural porous media.

Equation 4 is not in its most common form. Dividing through both sides by A and treating $\Delta P/L$ as a limit where L approaches some small differential distance allows equation 4 to be written in its more common form as

$$u_d = -\frac{\kappa}{\mu}\frac{dP}{dx}, \qquad \textbf{Eq. (5)}$$

known as Darcy's Law, where u_d is the Darcy velocity [m·s^{-1}], and can be expressed more generally as

$$\boldsymbol{u}_d = -\frac{\kappa}{\mu}(\boldsymbol{\nabla} P - \rho \boldsymbol{g}), \qquad \textbf{Eq. (6)}$$

to account for a 3-dimensional pressure gradient and gravity, $\boldsymbol{g}$ [m·s^{-2}]. Although $\boldsymbol{u_d}$ has units of velocity, it is important to remember that it was derived from and represents the volumetric fluid flux, the flow rate passing through the porous medium per unit of total cross sectional area of the domain. This facial area of porous media also includes the solid fraction of the material, and as a result, the Darcy velocity of the fluid is not the same as the local velocity of the fluid traveling through the pores, u_{pore} [m·s^{-1}]. On average, these two velocities are related to one another by the equation $u_d = \phi u_{pore}$. Furthermore, the ∇P term falsely implies that the limit of ΔL approaching zero was taken, when in fact, ΔL was collapsed insomuch that the control volume being considered remained large enough to include a representative number of pores such that the permeability remained independent of L, required by equation 4.

Despite these restrictions, the Darcy velocity provides a convenient way to describe the global velocity field and can be incorporated into other equations in a manner similar to the traditional fluid velocity vector. For example, considering a small element of porous media (still large compared to pore size), the governing equation for mass conservation of a single fluid within this control volume can be written as

$$\frac{\partial}{\partial t}[\phi\rho] + \boldsymbol{\nabla} \cdot (\rho \boldsymbol{u}_d) = 0 \qquad \textbf{Eq. (7)}$$

where ρ is the density of the fluid [kg·m^{-3}]. The porosity in the leftmost term is required because the fluid only occupies the empty space within the control volume while the fluid velocity (Darcy velocity) entering and leaving the control volume is averaged over the entire face of the domain. Similar to the traditional conservation of mass of a fluid, when the material derivative of density is equal to zero (fluid is incompressible) and the porosity is constant, equation 7 becomes the familiar $\boldsymbol{\nabla} \cdot \boldsymbol{u}_d = 0$.

In natural porous media, multiple fluids are often present within a single domain. For example, oil reservoirs almost always contain water and natural gas in addition to the mixture of liquid hydrocarbons (Land, 1968). In situations involving multiple fluids, additional parameters are required to describe the state of the domain. The most important parameter is saturation, S_i [-], which is defined as the volume of fluid i per volume of pore space. Using this definition, when only a single fluid is present within the domain the saturation of this fluid must be equal to 1. Similarly, if N fluids are present within the domain, then by definition

$$\sum_{i}^{N} S_i = 1\,. \qquad \textbf{Eq. (8)}$$

The conservation of mass for a single fluid phase (fluid *i*) in the presence of multiple fluids within the porous medium and in the absence of any chemical reactions or dissolution is given by

$$\frac{\partial}{\partial t}[\phi \rho_i S_i] + \boldsymbol{\nabla} \cdot (\rho_i \boldsymbol{u}_{di}) = 0 \qquad \textbf{Eq. (9)}$$

where the fluid Darcy velocity is slightly modified and given by

$$\boldsymbol{u}_{di} = \frac{-\kappa k_i}{\mu_i}(\boldsymbol{\nabla} P_i + \rho_i \boldsymbol{g}) \,. \qquad \textbf{Eq. (10)}$$

Because of the multiple fluids present, even if the fluid in question is incompressible, equation 9 tells us that $\boldsymbol{\nabla} \cdot \boldsymbol{u}_{di} \neq 0$ but rather $\boldsymbol{\nabla} \cdot \boldsymbol{u}_{di} = -\phi \partial S_i / \partial t$; the divergence of the velocity field is nonzero because the amount of fluid *i* within the control volume can still change by displacing the other fluids present. The Darcy velocity in equation 10 now includes an additional term called the relative permeability, k_i [-]. Because the permeability, κ, was derived assuming a single fluid was present, when multiple fluids occupy the pores, the observed flow rate of any one fluid is reduced relative to its flow rate when it occupies the pore space alone; the presence of additional fluids modifies the geometry of the pore network (and the boundary conditions) experienced by the fluid of interest. As a result, the relative permeability is defined to vary between 0 and 1, and is a function of saturation. Here again, no general equation can be derived to define k_i as a function of S_i for an arbitrary porous medium, because the effect of additional fluids on geometry and boundary conditions is hard to predict and can depend on history. However, many relative permeability models of the form $k_i = \mathrm{fn}(S_i)$ have

been proposed. These models all contain one or more fitting parameters whose values are adjusted to best-fit experimental data (Corey, 1954; Land, 1968; van Genuchten, 1980).

Finally, the presence of multiple fluid phases introduces another important phenomenon absent in single phase flow: capillary pressure (Leverett, 1941). Due to the surface tension that may exist at the interfaces of neighboring fluids, and the preferential wettability of the solid matrix to one or more fluid phases present, the pressures in neighboring fluids will differ and the pressure drop across the interface of these adjacent fluids is the capillary pressure, P_{cap} [Pa], broadly defined by the Young Laplace equation, given by

$$P_{cap} = P_{nw} - P_w = \sigma\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \qquad \textbf{Eq. (11)}$$

where P_{nw} is the pressure at the interface in the non-wetting fluid [Pa], P_w is the pressure at the interface in the wetting fluid [Pa], σ is the surface tension between the two fluids and the solid wall [N·m^{-1}], and R_1 and R_2 are the principle radii of curvature of the menisci that separate the fluids.

For simple geometries, the Young Laplace equation can often be simplified, for example, in a single, cylindrical pore occupied by two immiscible phases, the capillary pressure is given by

$$P_{cap} = \frac{2\sigma\cos\theta}{r_p} \qquad \textbf{Eq. (12)}$$

where r_p is the radius of the pore [m] and θ is the small angle between the solid wall and the curved interface between the two fluids [rad]. In natural porous media where

the internal geometry is often unknown, it is not possible to explicitly define the capillary pressure between fluid phases using such a pore-scale treatment. Instead, a global model is used to related the capillary pressure to the saturation of the wetting phase, given by

$$P_{cap}(S_w) = \left(\sigma\sqrt{\frac{\phi}{\kappa}}\right)\mathcal{J}(S_w) \qquad \textbf{Eq. (13)}$$

where $\mathcal{J}(S_w)$ is the Leverrett J-function [-], which is often a polynomial with fitting parameters to best-fit experimental data, similar to models of relative permeability (Leverett, 1941). The prefactor in equation 13 can be considered an average capillary pressure, where an effective pore radius is given by $\sqrt{\kappa}$. A brief history of the Leverett J-function is provided in Appendix 1, along with a summarized work of Noaman El-Khatib, who, in 1995, attempted to derive a universal relationship between permeability and the Leverett J-function; although, to-date, no such universal relationship has been shown to be valid.

The purpose of these models is to capture the pore-scale physics of multiphase flow (relative permeability and capillarity) in an averaged way that relates to the global fluid properties of saturation and the Darcy velocity, such that the governing conservation equation(s) can be solved and the velocity and saturation fields can be fully described. Altogether, the equations presented in this chapter provide an overview of how flows in porous media are characterized and modeled and will be necessary to understand the details of the research discussed in proceeding chapters.

1.3 Introduction to carbon sequestration

Carbon sequestration has been an active topic of research in the fields of engineering, geology, social science, and public policy for more than a decade (Figueroa et al., 2008; Yang et al., 2008). The concept of carbon sequestration is to capture carbon dioxide and inject it as a fluid deep underground, where it could remain indefinitely, while the underlying purpose of sequestering carbon is to reduce the concentration of CO2 within the earth's atmosphere that is currently contributing to global warming (DOE, 2012). Because much is known about injecting fluids into the subsurface by the oil and gas industry, many of the most difficult challenges of carbon sequestration lay in the realms of public policy and the federal government. Although an analysis of these issues is outside the scope of this dissertation, it is worth noting that these challenges include responses to the following questions: (1) should carbon capture be mandated, (2) how should carbon capture be financed, (3) where will storing CO2 underground cause the least amount of public outcry, and (4) what are the consequences of CO2 returning to the surface and who should be responsible if a sequestration site fails (Grainger and Kolstad, 2010; Rodosta et al., 2014; Zunsheng et al., 2014).

In addition to these public policy questions, there also remain many engineering challenges to carbon sequestration. Due to the high cost of drilling and obtaining subsurface data, there is large uncertainty of the permeability and porosity of any reservoir (Jordan and Doughty, 2009; Khaninezhad et al., 2012; Li and Jafarpour, 2010). Modeling large reservoirs often requires parameter studies to provide a range of outcomes and even after doing so, flow fields may be dominated by unforeseen faults

within the domain (Bejan, 2006; NETL and DOE, 2010). It can be also difficult to predict the long-term fate of the CO2 and the effects on the reservoir (Leonenko and Keith, 2008; Li and Jafarpour, 2010). Since the CO2 is usually less dense than the native fluids, there is a tendency for CO2 to rise within the domain and depending on the topography may spread great distances, potentially increasing the risk of leaking to the surface (Huber et al., 2016; Kopp et al., 2010). Depending on the native fluids within the reservoir, CO2 may also dissolve and diffuse into the surrounding fluids. The work in this dissertation analyzes three particular aspects of carbon sequestration relevant to these issues and inherent uncertainties in modeling the dynamics of fluids in the subsurface: (1) an injection strategy to rapidly immobilize the CO2 after it was been pumped into the reservoir, (2) the propensity for CO2 to become remobilized and the behavior and motion after remobilization has occurred, and (3) the dynamics of mass transfer during caprock dry-out in the presence of cavitation events. The goal of this work is to provide insight into the important physics of these processes and to present practical solution techniques for each topic and the current results of each analysis

CHAPTER 2

ANALYSIS OF A TIME DEPENDENT INJECTION STRATEGY TO ACCELERATE THE RESIDUAL TRAPPING OF SEQUESTERED CO_2 IN THE GEOLOGIC SUBSURFACE

2.1 Introduction

Carbon sequestration has been proposed as one possible strategy to mitigate anthropogenic carbon emissions; and as a collective technology requires the capture, transport, and subsurface injection of CO_2 originating from the flue gas of power plants. Although this strategy has many barriers to implementation, such as high capital investment requirements and federal regulatory hurdles, geologic surveys within North America suggest enormous potential for the subsurface to retain CO_2. For example, the continental US has an estimated 2,000 – 20,000 $GtCO_2$ storage capacity in the geologic subsurface, while it currently emits ca. 9 $GtCO_2$ annually into the atmosphere (Dooley et al., 2005; Finley, 2005; Meyer, 2007; NETL and DOE, 2010; Rodosta et al., 2014; Steadman et al., 2006). The vast majority of this estimated storage capacity (> 90%) is within saline aquifers, reservoirs which already contain brine. In recent years, several pilot-scale injection facilities have been built in the United States and China to demonstrate sequestration and to identify the technical limitations of operation and scale-up. Examples include the Plant Barry CCS project in Mobile, Alabama and the Ordos CCS/EOR project in Inner Mongolia, China (Zunsheng et al., 2014). In addition to the challenges associated with optimizing CO_2 capture from flue gas sources, it is often difficult to predict the exact behavior of mobile CO_2 within the subsurface both

during the injection process and after the injection well has been decommissioned. Ultimately, limiting post-injection CO_2 migration is critical since the success or failure of sequestration hinges upon preventing CO_2 from migrating back to the surface and reentering the atmosphere. In this chapter, we propose and analyze a time-dependent CO_2-brine injection strategy aimed to rapidly immobilize CO_2 compared to conventional CO_2 injection strategies used today. Our analysis presents the equations which govern the dynamics of CO_2 in the presence of brine within the geologic reservoir and develops a solution technique based on the method of characteristics to understand and model the dynamics of this injection process.

Conventional geologic carbon sequestration, represented in the schematic diagram in Figure 2a (Doughty, 2010), uses a single injection well drilled to a depth such that the ambient pressures and temperatures at the bottom hole within the reservoir are high enough to keep the CO_2 in the supercritical phase (generally a depth greater than 800 m). Supercritical CO_2 is injected for some fixed period of time into the desired rock formation. These subsurface rock formations (defined as zones of similar structure and composition) typically extend laterally tens to hundreds of kilometers, while vertically they may only extend tens of meters or less (DOE, 2012). This stratified topology of alternating layers of high and low permeability rock causes the injected CO_2 to preferentially convect radially outward from the injection site into the zones of high permeability, while the layers of lower permeability impede vertical migration from occurring (Figure 2a). Usually, CO_2 is injected into many high-permeable layers simultaneously, and the extent of the plume grows throughout the injection process until the well is decommissioned.

After injection ends, the CO_2 continues to move in the subsurface. This plume motion is generally driven by buoyancy differences between the CO_2 and the native brine within the reservoir. Over time, plume motion slows and eventually ceases due to a variety of phenomena. These phenomena, which promote plume immobilization, are broadly referred to as trapping mechanisms and have been categorized into four broad groups: structural, residual, solubility, and mineral. A full review of these mechanisms in the context of carbon sequestration has recently been discussed by (Huppert and Neufeld, 2014), but in brief: structural trapping requires a spatial variation in pore structure which results in an abrupt decrease in permeability and typically occurs at the boundaries of adjacent rock formations. Caprock seals located above a reservoir are the most prominent example of structural trapping, and are largely responsible for preventing injected CO_2 from immediately returning to the surface. Residual trapping occurs when isolated pockets of CO_2 are immobilized in pore spaces by capillary forces, and is a result of bulk CO_2 draining or receding within a porous medium in the presence of at least one other fluid phase. Finally, solubility and mineral trapping are possible depending on the specific properties of the reservoir and refer to the dissolution and chemical reaction of CO_2 with the native fluids and solid matrix, respectively.

Although all of these trapping mechanisms may occur within a given reservoir, not all of these mechanisms contribute significantly to plume immobilization within time scales comparable to injection times or, arguably, over the duration of site stewardship. Immediately following injection (Figure 2b(i)), the majority of CO_2 is mobile and upward mobility is prevented by structural caprock. As the plume rises and thins along the caprock (Figure 2b(ii)), residual CO_2 remains in the trailing edges of the

plume while, globally, the plume volume also decreases due to local dissolution of CO_2 into the surrounding brine. Eventually, the CO_2 at the leading edge of the plume is unable to overcome the capillary forces of the CO_2-brine-rock contact line and the entire plume becomes immobile, while dissolution continues to occur at the boundaries of the plume. In the absence of chemically reactive geology, when CO_2 is injected in brine aquifers, the end result is the complete dissolution of the CO_2. This final state has been observed in naturally trapped pockets of CO_2 within the subsurface as well as predicted by existing models of sequestration (Ennis-King and Paterson, 2007; Gilfillan et al., 2009). However, dissolution slows considerably after plume immobilization because local convection at the boundary between the CO_2 plume and the native brine is driven by the small density difference between brine saturated with CO_2 near the boundary and brine undersaturated with CO_2 below the boundary. The time scales for mass transfer to occur by dissolution in these scenarios have been shown to be $O(10^6 – 10^9$ years) (Ennis-King and Paterson, 2007).

On shorter time scales, structural and residual trapping are the dominate mechanisms contributing to plume immobilization and reservoir security. In an effort to predict immobilization times, numerous studies have been published that model the dynamics of post-injection plume motion both with and without the presence of residual trapping (Golding et al., 2011; Hesse et al., 2008, 2007) and in specific geologic locations using in-situ topography and heterogeneity (Doughty, 2010; Hesse et al., 2008; Juanes et al., 2010). These models estimate that injected CO_2 remains mobile on time scales of order 100 – 10,000 years, depending on the properties of the rock, the geometry of the reservoir, and the volume of injected CO_2.

Several authors have considered using brine injection, brine extraction, multiwell CO_2 injection, or surface mixing to reduce the time required to immobilize the CO_2. Although surface mixing of brine and CO_2 offers the fastest mechanism to complete dissolution, it requires additional capital investment for surface equipment and has been shown to impair injectivity during pumping (Delshad et al., 2010). Restricting possibilities to within the subsurface, Shamshiri and Jafarpour investigated a sequestration strategy using four injection wells with the goal of alternating the injection location and volume to increase the spatial extent of the CO_2 plume and increase the mass fraction of residually trapped gas (Shamshiri and Jafarpour, 2012). They found that nearly half of the CO_2 could be residually trapped within 300 years after injection ended. The addition of wells dedicated solely to brine extraction was considered by Buscheck, who modeled a system of nine extraction wells oriented in two concentric rings around the CO_2 injection well (Buscheck et al., 2012). By alternating the volume and location of brine extraction, the pressure buildup within the CO_2 plume was reduced; this reduction in pressure would in turn reduce CO_2 plume motion after CO_2 injection ceased. However, this strategy required that produced brine not be reinjected back into the formation and alternate uses for the extracted brine would need to be found. Finally, Leonenko and Keith modeled extracting brine from one reservoir location and reinjecting the brine at the CO_2 injection well in order to reduce pressure buildup during the injection of CO_2 (Leonenko and Keith, 2008). For the particular "top-hat" reservoir geometry that they analyzed, more than 75% of the CO_2 was immobilized by dissolution after 300 years.

In this chapter, we propose and analyze a specific sequestration strategy using a single injection well (Figure 2c). Following an injection of CO_2, reservoir brine from elsewhere in the formation is extracted and pumped back into the formation at the original injection site. This brine flooding helps to immobilize the sequestered CO_2 plume by residual trapping mechanisms on time scales of the order 1 – 5 times the original CO_2 pumping time (Figure 2d), after which, additional brine may be injected to achieve complete dissolution. Starting from general mass conservation equations for brine and CO_2, we derive a set of coupled equations to describe the evolution of CO_2 saturation and dissolved CO_2 within the reservoir, assuming one dimensional, pressure driven, incompressible, two-phase flow. These equations are used to estimate the effectiveness of brine injection as a means of promoting the residual trapping of CO_2 and are then solved with the method of characteristics to model the dynamics of brine flooding. Finally, we conclude with 3D axisymmetric finite volume numerical simulations using TOUGH2, which allow additional considerations such as gravity and fluid compressibility to be included in the model. From these results, we conclude that the simple, one-dimensional flow model is sufficient to describe the dynamics of CO_2 residual trapping and that this trapping mechanism can reduce the mobile fraction of CO_2 within the plume to less than 10 % using brine volumes of the order of the volume of CO_2 initially injected.

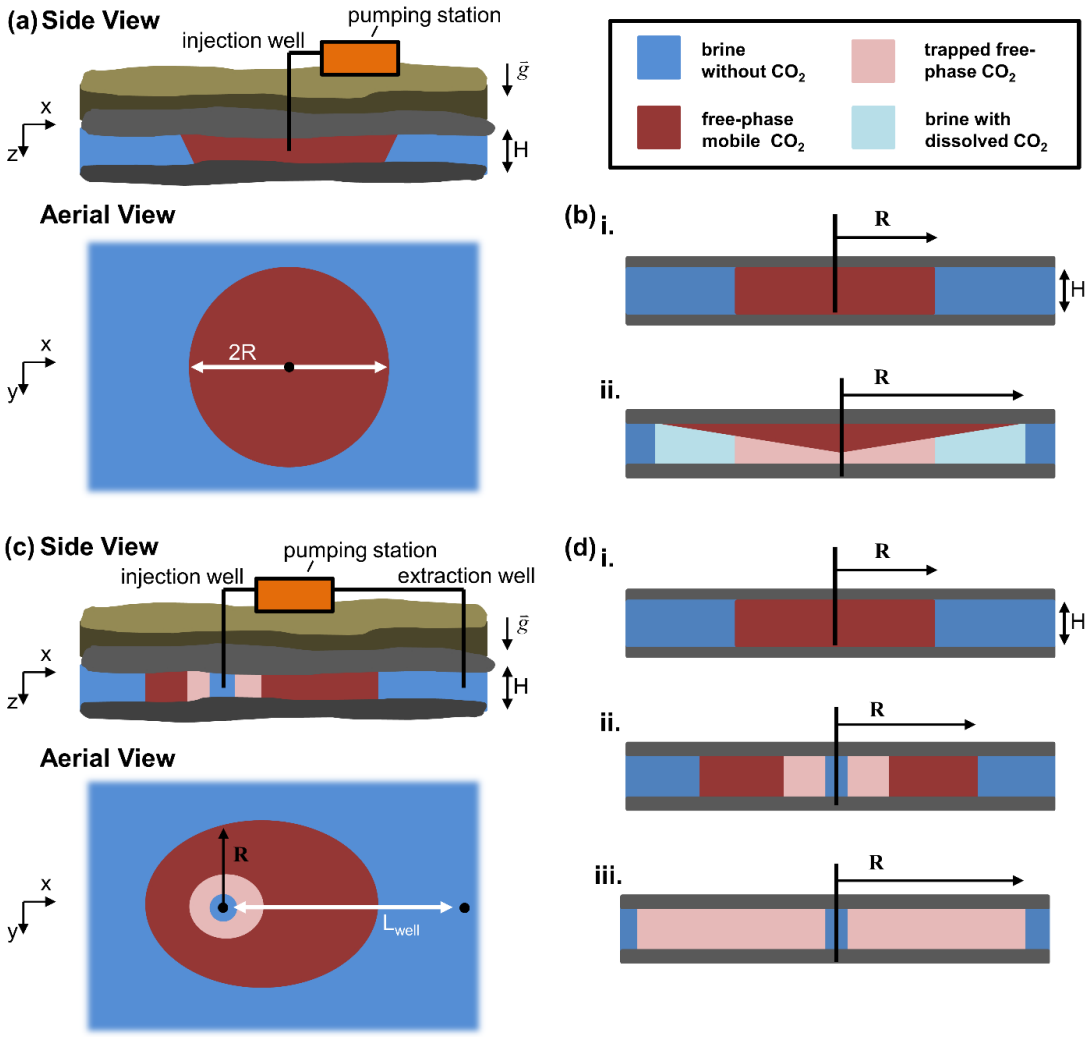


Figure 2. (a) Conventional CO_2 injection into a brine aquifer using a single well. (b) (i) During CO_2 injection, the plume will spread radially outward, displacing the brine. (ii) After injection ends, the plume will rise and spread along the caprock leaving behind residually trapped pockets of CO_2. Concurrently, some of the CO_2 will dissolve into the surrounding brine. (c) Our proposed injection strategy prescribing that brine be injected after the CO_2. If the flow field is primarily unidirectional, the brine filled rock will imbibe into pores containing CO_2 as shown in (d) (i) – (iii), effectively immobilizing the plume by residual trapping. Additional brine can be injected to completely dissolve the immobile CO_2.

2.2 Analysis

In this section, we present the nonlinear partial differential equations that govern the dynamics of CO_2 injection in the presence of brine. Without simplifying assumptions, these equations may only be solved numerically. However, as we will demonstrate, several reasonable assumptions can be made that will reduce the complexity of the equations to a point where analytic solutions can be discussed. The result is a lower order model that accurately captures the important dynamics of the injection process and provides useful insight into the conditions required for brine flooding to be effective at immobilizing CO_2.

2.2.1 Deriving the governing equations

As mentioned in section 2.1, the majority of natural reservoirs that could be used for CO_2 storage are saline aquifers, meaning that they already contain brine (solution of water and dissolved salts). Although, in general, a reservoir may contain many native fluids, we will limit the scope of our analysis to that of a reservoir initially containing only brine. In the absence of chemical reactions, injected CO_2 will first exist as a supercritical fluid phase (SCFP), but it may also dissolve into the surrounding brine; we will denote the dissolved CO_2 as the aqueous phase. In this analysis, we will neglect all CO_2-salt interactions as well as the dissolution of any species into the SCFP CO_2. Therefore, the components being considered are: SCFP CO_2, aqueous CO_2, and brine. To begin, the conservation of mass for the SCFP CO_2 within the reservoir can be written as

$$\frac{\partial}{\partial t}[\phi \rho_c S_c] + \nabla \cdot (\rho_c \boldsymbol{u}_{dc}) = -h\phi S_b \left(C_{aq}^{+} - C_{aq}\right), \qquad \textbf{Eq. (14)}$$

where ϕ is the porosity of the porous medium [-], ρ_c is the density of SCFP CO_2 [kg·m^{-3}], S_c is the saturation of CO_2 [-], $\boldsymbol{u}_{dc}$ is the Darcy velocity of CO_2 [m·s^{-1}], h is the mass transfer capacity coefficient [s^{-1}], S_b is the saturation of brine [-], C_{aq} is the concentration of CO_2 dissolved in the brine [kg CO_2·m^{-3} brine], and C_{aq}^{+} is the maximum concentration of CO_2 that can be dissolved in the brine (based on reservoir temperature and pressure) [kg CO_2·m^{-3} brine]. The right hand side of equation 14 accounts for the local dissolution of SCFP CO_2 into the surrounding brine phase. We note that mass transfer coefficients, h_m, are often defined as the mass transferred per unit area per concentration difference per second [$\mathrm{kg \cdot m^{-2} s^{-1} \Delta C^{-1}}$], implying that the right hand side of equation 14 would be written as $-h_m A\phi S_b \left(C_{aq}^{+} - C_{aq}\right)$, where A is the interfacial area between the SCFP CO_2 and brine. However, since the available contact area between the two fluid phases is not usually known locally within the porous media, we have lumped $h_m A$ into the mass transfer capacity coefficient, h. Similarly, the conservation of mass for the brine within the reservoir can be written a

$$\frac{\partial}{\partial t}[\phi \rho_b S_b] + \nabla \cdot (\rho_b \boldsymbol{u}_{db}) = 0, \qquad \textbf{Eq. (15)}$$

where ρ_b is the density of brine [kg·m^{-3}] and $\boldsymbol{u}_{db}$ is the Darcy velocity of brine [m·s^{-1}]. Changes in brine density as a result of the presence of aqueous CO_2 have been neglected, as these changes are small relative to the density of brine unsaturated with CO_2. Finally, the conservation of mass for the CO_2 dissolved in brine can be written as

$$\frac{\partial}{\partial t}\left[\phi S_b C_{aq}\right] + \boldsymbol{\nabla} \cdot \left(C_{aq}\boldsymbol{u}_{db}\right) = h\phi S_b\left(C_{aq}^{+} - C_{aq}\right). \qquad \textbf{Eq. (16)}$$

In equations 14 - 16, the saturation of phase 'i', S_i, within a control volume is defined as the volume of phase 'i' present in the control volume divided by the total pore space of the control volume. Here the subscript 'i' is replaced with either 'c' or 'b' to denote SCFP CO_2 or brine, respectively. From this definition, the relationship between the CO_2 saturation and brine saturation can be expressed as

$$S_c + S_b = 1 \qquad \textbf{Eq. (17)}$$

since we have assumed that CO_2 and brine are the only two fluid phases present in the porous medium.

In multiphase flow, the Darcy velocity of fluid phase 'i', $\boldsymbol{u}_{di}$ [m·s^{-1}], can be written as

$$\boldsymbol{u}_{di} = \frac{k_i \kappa}{\mu_i}\left(-\boldsymbol{\nabla} P_i + \rho_i \boldsymbol{g}\right), \qquad \textbf{Eq. (18)}$$

where κ is the intrinsic permeability of the porous medium [m^2], μ_i is the viscosity of the fluid [Pa·s], k_i is the relative permeability of the fluid in phase *i* in the porous media [-], P_i is the fluid pressure [Pa], and $\boldsymbol{g}$ is the acceleration due to gravity. In general, equations 14 - 18 can be solved by coupling these equations to the pressure field within the reservoir, the relative permeability of each fluid phase, and the thermodynamic equations of state for each species.

In our proposed injection strategy, CO_2 is pumped into a reservoir for some fixed period of time. When CO_2 injection ends, brine is then pumped into the reservoir using

the same injection well. This brine may be harvested from a production well within the same geologic formation, provided the two well sites are drilled far enough apart to prevent the production of CO_2 (Figure 2c). We assume that the CO_2 will be sequestered in reservoirs deeper than 2 km, where the density of CO_2 is >85% that of brine. Storing the CO_2 at these high densities makes better use of the storage volume available in the reservoir and limits buoyancy forces between the two phases. At these depths, the reservoir pressure is approximately 20 MPa, corresponding to a formation in which the native brine is at hydrostatic equilibrium, and over the range of pressures observed during the injection process (20 – 40 MPa), the density of CO_2 is relatively constant, see Figure 3.

To obtain a solution using the method of characteristics, we will make a few assumptions about the multiphase flow. First, we will assume that the intrinsic permeability of the porous medium and fluid viscosity are homogenous and isotropic. Second, we assume that the buoyancy driven migration of CO_2 in the z direction is negligible throughout the injection process. This assumption is valid provided that the time scale for CO_2 migration in the z direction, τ_{cz}, is much larger than the time scale for CO_2 convective transport in the r direction, τ_{cr}. The time scale for the buoyancy driven vertical rise of CO_2 toward the top of the permeability layer is given by $\tau_{cz} \sim \mu_b H/(\kappa \Delta\rho g)$, where $\Delta\rho$ is the density difference between brine and CO_2 [kg·m^{-3}] driving the flow. Here, the viscosity of brine, μ_b, is used because in the confined countercurrent arrangement of CO_2 rising while brine is sinking, the resistance to fluid motion will primarily come from the phase with the larger viscosity. The time scale for convective CO_2 transfer in the radial direction is given by $\tau_{cr} \sim 2\pi H \rho_c R^2/\dot{m}_c$, where R

is the estimated extent of the plume [m] and $\dot{m}_c$ is the mass flow rate of CO_2 into the injection well [kg·s^{-1}]. R may be estimated by $[(\dot{m}_c \Delta t)/(\phi S_c^+ \pi H \rho_c)]^{1/2}$. Using characteristic values of $\mu_b = 1 \times 10^{-3}\ Pa \cdot s$, $\kappa = 1 \times 10^{-14}\ m^2$, and the parameters listed in Table 1, $R \sim 85\ m$ and $\tau_{cz}/\tau_{cr} \approx 5$. Third, we assume that the capillary pressure, P_{cap} [Pa], can be neglected throughout the injection process; this assumption is valid when the working pressures of the system, ΔP_{pump} [Pa], are much greater than the capillary pressure between the two phases. The working pressure may be estimated by $\Delta P_{pump} \sim \dot{m}_c \mu_b/(2\pi H \rho_c \kappa)$, which from the parameters listed in Table 1, $\Delta P_{pump} \approx 5.6\ MPa$. The capillary pressure may be estimated by $P_{cap} \sim 2\sigma/\sqrt{\kappa}$, where σ is the interfacial surface tension between the supercritical CO_2 and brine. At reservoir conditions, it has been experimentally demonstrated (Bennion and Bachu, 2006a) and predicted by molecular simulation (Lafitte et al., 2010) that $\sigma \approx 20\ mN/m$ at 20 MPa. Using this result, $P_{cap} \approx 400\ kPa$, which is significantly less than ΔP_{pump}. Finally, we assume that the brine and CO_2 are incompressible. The CO_2 can be assumed incompressible if $\Delta\rho_c/\rho_{co} \ll 1$, where ρ_{co} is the initial density of CO_2 at the injection well and $\Delta\rho_c$ is the change in CO_2 density as a result of the pressure gradient required to pump the fluid into the reservoir. From the third assumption above, this pressure gradient is given by ΔP_{pump}, and using the initial reservoir pressure, P_o, of 20 MPa, $\Delta\rho_c \approx 50$ kg/m^3, from the data in Figure 3. Using this result, $\Delta\rho_c/\rho_{co} \approx 0.06$. Although not shown here, justifying the assumption that the brine is also incompressible can be done in a similar fashion.

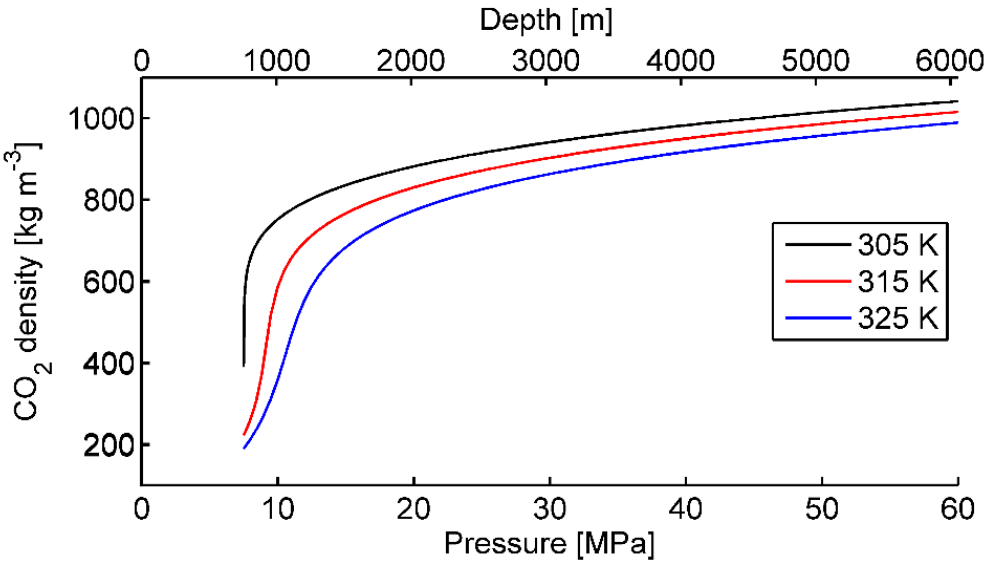


Figure 3. Density of supercritical CO_2 as a function of pressure for isothermal conditions at temperatures of 305 K, 315 K, and 325 K. Depth values on the upper x-axis correspond to hydrostatic pressure. Density vs. pressure data comes from the National Institute of Standards and Technology (NIST) (Linstrom and Mallard, 2011).

After applying these assumptions, equations 14 - 16 become one dimensional in r and can be written as

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi r}\frac{\partial}{\partial r}(r u_{dc}) = \frac{-hS_b\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c} \qquad \textbf{Eq. (19)}$$

$$\frac{\partial S_b}{\partial t} + \frac{1}{\phi r}\frac{\partial}{\partial r}(r u_{db}) = 0 \qquad \textbf{Eq. (20)}$$

$$\frac{\partial C_{aq}}{\partial t} + \frac{u_{db}}{\phi S_b}\frac{\partial C_{aq}}{\partial r} = h\left(C_{aq}^{+} - C_{aq}\right), \qquad \textbf{Eq. (21)}$$

where the Darcy velocity for each phase is now given by

$$u_{di} = \frac{-k_i \kappa}{\mu_i}\frac{\partial P}{\partial r} \qquad \textbf{Eq. (22)}$$

since the pressure gradient in each phase is assumed to be the same. To solve for the pressure gradient, we substitute equation 22 into equations 19 and 20 and add them together. The combined expression can now be simplified by recognizing that $\partial(S_c + S_b)/\partial t = 0$, from equation 17. After making this simplification, the solution to the pressure gradient can be written as

$$\frac{dP}{dr} = \frac{-1}{\left[\frac{k_c}{\mu_c} + \frac{k_b}{\mu_b}\right]\kappa} * \left[\frac{Q_{tot}}{2\pi Hr} + \frac{1}{r}\int_{r_o}^{r} \frac{-h\phi(1 - S_c)\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c} r' dr'\right], \qquad \textbf{Eq. (23)}$$

where Q_{tot} is the total volumetric flow rate entering the reservoir [$m^3 \cdot s^{-1}$], H is the height of the reservoir [m], r is the radial distance from the injection well [m], and r_o is a radial distance arbitrarily close to the injection well [m]. $r_o \neq 0$ because the pressure field diverges at the origin. The term $Q_{tot}/(2\pi Hr)$ arises as a result of satisfying the flux boundary condition imposed at the well ($r = r_o$), and replaces the term containing the arbitrary constant (of the form a/r) that would appear in the generalized version of equation 23. The integral on the right hand side of equation 23 describes the decrease in the volumetric flow rate arising from the loss of total fluid volume as SCFP CO_2 dissolves into the brine without significantly altering the density of the brine phase. Substituting equation 23 into equation 22, the Darcy velocities for CO_2 and brine can be written as

$$u_{dc} = \left[\frac{1}{1 + \eta f}\right] * \left[\frac{Q_{tot}}{2\pi Hr} + \frac{1}{r}\int_{r_o}^{r} \frac{-h\phi(1 - S_c)\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c} r' dr'\right] \qquad \textbf{Eq. (24)}$$

and

$$u_{db} = \left[1 - \frac{1}{1+\eta f}\right] * \left[\frac{Q_{tot}}{2\pi H r} + \frac{1}{r}\int_{r_o}^{r} \frac{-h\phi(1-S_c)\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c} r' dr'\right], \quad \textbf{Eq. (25)}$$

respectively, where η is the ratio of the fluid viscosities (μ_c/μ_b) [-] and f is the ratio of the relative permeabilities (k_b/k_c) [-].

It is often convenient to lump the viscosity, intrinsic permeability and relative permeability terms into a single coefficient, which is usually called the phase mobility or mobility ratio in the literature (Hesse et al., 2007; Pegler et al., 2014; Vilarrasa et al., 2010). However, here we maintain the segregation of η and f because (1) throughout the injection process we assume that the ratio of the fluid viscosities remains constant, and (2) although CO_2 sequestration is likely to occur at similar depths, temperatures, and pressures, significant variability exists in the relative permeability properties of the CO_2 and brine due to geologic diversity; therefore the above formulation makes it easier to highlight the effects of such variability in the context of the proposed sequestration strategy.

In general, f cannot be analytically derived due to its complex dependencies on the porous media and the fluid phases present (Bachu, 2013). However, it has been shown that to a close approximation, a modified Brooks-Corey empirical model can be used to describe the relative permeability behavior of each phase. These relative permeability functions can be written as

$$k_c = k_c^{+}\left(\frac{S_c - S_c^{-}}{S_c^{+} - S_c^{-}}\right)^n \quad \textbf{Eq. (26)}$$

and

$$k_b = k_b^+ \left(\frac{S_c^+ - S_c}{S_c^+ - S_c^-} \right)^m, \qquad \textbf{Eq. (27)}$$

where S_c^- is the residual (minimum) saturation of CO_2 [-], S_c^+ is the maximum saturation of CO_2 (corresponding to the minimum brine saturation) [-], k_c^+ is the maximum relative permeability of CO_2 [-], and k_b^+ is the maximum relative permeability of brine [-] (Alpak et al., 1999; Bachu, 2013; Müller, 2010). The Corey exponents, *n* and *m*, are independent of one another and have been shown to vary between the values of 1 and 7 (Bachu, 2013). These exponents affect the curvature of the relative permeability functions, as shown in Figure 4b, while the endpoints of the relative permeability functions are set by S_c^-, S_c^+, and k_i^+. The residual saturation refers to the volume of the receding fluid phase which remains trapped in the porous medium after an invading fluid has filled the active pathways within the porous network. In this context, we refer to the process of CO_2 injection followed by imbibition of brine (Figure 4a). The saturation values of CO_2 after injection (Figure 4a(ii)) and after complete brine flooding (Figure 4a(iii)) correspond to the maximum and minimum saturations, S_c^+ and S_c^-, respectively. Experiments using sandstone cores revealed that maximum CO_2 relative permeability, k_c^+, can vary between widely between 0.05 and 1 (Bachu, 2013; Bennion et al., 2008; Burnside and Naylor, 2014). We also note, that while equations 26 and 27 do not include the hysteretic effects that may be associated with multiple drainage and imbibition cycles (Juanes et al., 2006), these equations provide the simplest framework to capture the physics of residual trapping in the context of a single CO2 injection followed by a single brine injection.

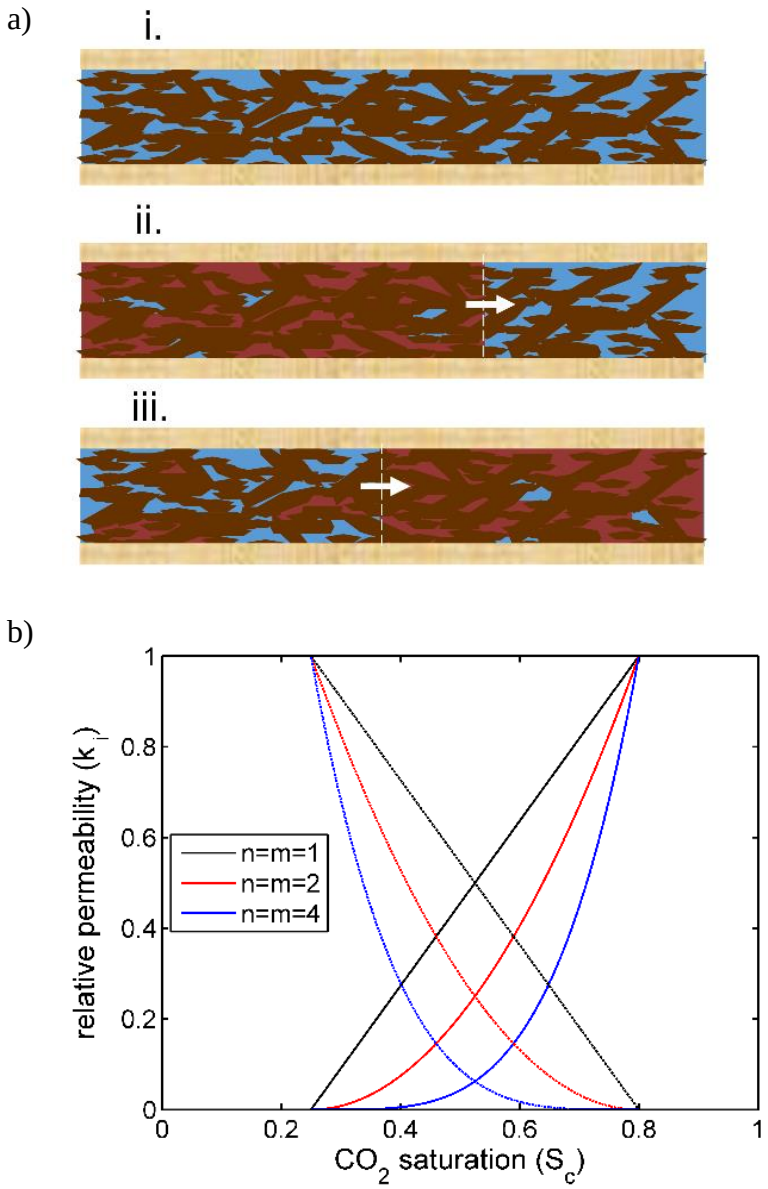


Figure 4. (a) (i) Prior to CO_2 injection, the pore spaces of the reservoir are saturated with brine. (ii) During injection, CO_2 invades the pores and displaces some of the brine. The CO_2 saturation increases until reaching a maximum value (S_c^+), at which point the saturation of brine is irreducible. (iii) After CO_2 injection ends, brine is injected, reducing the CO_2 saturation until the minimum residual saturation is reached (S_c^-). (b) Relative permeability curves for CO_2 (solid lines) and brine (dashed lines) are plotted for three different values of Corey exponents using the parametric values in Table 1 and equations 26 and 27.

mass flow rate of injected fluids (CO_2 or brine) into the reservoir, $\dot{m}_i$ [kg·s^{-1}]	3	duration of CO_2 injection [days]	90
maximum CO_2 saturation, S_c^+ [-]	0.80	density of CO_2, ρ_c [kg·m^{-3}]	850
minimum CO_2 saturation, S_c^- [-]	0.25	density of brine, ρ_b [kg·m^{-3}]	1010
maximum relative permeability of CO_2, k_c^+ [-]	1.0	C_{aq}^+ [kg CO_2·m^{-3} brine]	60
maximum relative permeability of brine, k_b^+ [-]	1.0	reservoir height, H [m]	10
ratio of μ_c/μ_b, η [-]	0.1	ratio of C_{aq}^+/ρ_c, ψ^+ [-]	0.07
porosity of reservoir, ϕ [-]	0.15	mass transfer capacity coefficient, h [s^{-1}]	1 x 10^{-4}

Table 1. The parameter values for CO_2, brine, the porous matrix, and the pumping parameters used for the analysis in this chapter, unless otherwise noted.

2.2.2 Estimating brine volume requirements and CO_2 footprint size

Before solving the governing equations in section 2.3, two important attributes of our proposed injection strategy, namely, brine volume requirements and CO_2 plume footprint size, may be estimated assuming a radially symmetric, homogeneous reservoir and using the representative values for CO_2, brine, and the porous matrix listed in Table 1. As depicted in Figure 2d, at the end of CO_2 injection, Figure 2d(i), brine is injected into the same well until the CO_2 is either fully trapped at the minimum residual saturation, represented in Figure 2d(iii), or is completely dissolved. The estimated volume of injected brine required to achieve complete residual trapping or complete dissolution will be a function of the minimum and maximum residual saturation values of CO_2, S_c^- and S_c^+, the ratio of the maximum CO_2 mass concentration in the brine to the SCFP CO_2 density, ψ^+, and the original volume of CO_2 injected into the reservoir. Using these characteristic properties of the porous medium and native brine, the

injection volume ratio brine:CO_2, γ [-], for the two desired reservoir end states can be expressed as equations 29 - 31, listed in Table 2.

To understand how these estimates are calculated, let us derive the first ratio listed, equation 29, in the absence of dissolution, meaning $\psi^+ = 0$. After the CO_2 has been injected into the reservoir the CO_2 volume can be estimated as $V_c = \phi\pi R_i^2 H S_c^+$, where R_i is the initial radius of the plume. After brine is injected, the CO_2 volume can be written as $V_c = \phi\pi R_f^2 H S_c^-$, since we now assume that the CO_2 has been spread throughout the reservoir until it is at the minimum saturation. The volume of brine required to accomplish this change can be estimated as the volume of brine within the final plume, set by R_f, less the brine volume originally found in the plume at R_i. Therefore, the volume ratio can then be written as

$$\frac{V_b}{V_c} = \gamma = \frac{R_f^2(1 - S_c^-) - R_i^2(1 - S_c^+)}{R_i^2 S_c^+} = \frac{S_c^+ - S_c^-}{S_c^+ S_c^-}, \qquad \textbf{Eq. (28)}$$

where $R_f^2 = R_i^2 S_c^+ / S_c^-$ from mass conservation of the CO_2 phase. The other volume ratio estimates listed in Table 2 can be found in a similar fashion (for further details, see Appendix 2).

In the volume estimates for complete dissolution, the difference in the ratio of brine:CO_2, γ, between the first injection cycle and any following injection cycles comes from the fact that we assume the reservoir is initially filled with brine, $S_b = 1$, and contains no dissolved CO_2. Consequently, the native brine will aide in partially dissolving the very first injection of CO_2. However, all following CO_2 injections will contact previously injected brine which is already saturated with dissolved CO_2. Using

the values listed in Table 1, we estimate that trapping can be achieved with $\gamma \approx 2.2$ and that complete dissolution can be achieved with $\gamma \approx 14$. However, in the absence of dissolution, the estimated volume ratio to achieve trapping increases to 2.75.

	immobilizing CO_2 through residual trapping	
All Injection Cycles	$\gamma = \left[\frac{S_c^+ - S_c^-}{(S_c^- + (1 - S_c^-)\psi^+)(S_c^+ + (1 - S_c^+)\psi^+)}\right]$ $\xrightarrow{\lim \psi^+ \to 0} \gamma = \left[\frac{S_c^+ - S_c^-}{S_c^- S_c^+}\right]$	**Eq. (29)**
	complete dissolution of the CO_2 into the reservoir brine	
First Injection Cycle	$\gamma = \left[\frac{S_c^+}{\psi^+(S_c^+ + (1 - S_c^+)\psi^+)}\right]$	**Eq. (30)**
All Following Injection Cycles	$\gamma = \frac{1}{\psi^+}$	**Eq. (31)**

Table 2. Summary of the estimated volume ratio (brine:CO_2), denoted as (γ), required to achieve either complete CO_2 immobilization by residual trapping mechanisms or complete CO_2 dissolution into the brine. Here, one injection cycle refers to one continuous injection of CO_2 followed by one continuous injection of brine. (See Appendix 2 for equation derivation.)

Because this injection strategy works by using brine flooding to dilute the CO_2 on shorter time scales than would occur naturally, the CO_2 footprint at the end of any flooding stage must be larger than the original CO_2 footprint, corresponding to a conventional injection strategy, represented in Figure 2d(iii) and Figure 2b(i), respectively. From geometry and the properties of the porous matrix, the ratio of the radial CO_2 footprint at the end of brine flooding to the radial CO_2 footprint without brine

flooding, for the cases of fully trapping and fully dissolving the CO_2 plume, may be estimated by

$$\frac{A_{trap}}{A_{orig}} = \frac{S_c^+}{S_c^-} \qquad \textbf{Eq. (32)}$$

and

$$\frac{A_{diss}}{A_{orig}} = \frac{S_c^+}{\psi^+}. \qquad \textbf{Eq. (33)}$$

Again using the values in Table 1, brine flooding would increase the CO_2 footprint by a factor of approximately 3.2 and 11.3 for complete residual trapping and complete dissolution, respectively.

2.2.3 Applying the method of characteristics to the governing equations

To provide insights into the dynamics of brine flooding, we now return to the governing equations from section 2.2.1. After substituting the Darcy velocity results from equations 24 - 25, it can be shown that equations 19 and 21 can be written as

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\left(\frac{Q_{tot}}{2\pi H r} + \frac{1}{r}\int_{r_o}^{r} -r' D dr'\right)\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right]\frac{\partial S_c}{\partial r} = \frac{D}{\phi}\left(\frac{1}{1+\eta f} - 1\right) \qquad \textbf{Eq. (34)}$$

and

$$\frac{\partial C_{aq}}{\partial t} + \frac{u_{db}}{\phi(1-S_c)}\frac{\partial C_{aq}}{\partial r} = h\left(C_{aq}^+ - C_{aq}\right), \qquad \textbf{Eq. (35)}$$

respectively, where $D = h\phi(1 - S_c)\left(C_{aq}^{+} - C_{aq}\right)/\rho_c$ (see Appendix 3 for derivation details). Equations 34 and 35 are a coupled set of wave equations, which are typically solved with the method of characteristics. Wave equations are a common class of PDEs that often arise in describing multiphase flow in porous media, e.g. Buckley-Leverett equations (Fokas and Yortsos, 1982; Marchesin and Plohr, 2001; Silin et al., 2009), and the method of characteristics have been used to solve for the flow of compressible CO_2 and brine in the absence of dissolution (Moghanloo, 2012). The solution to equations 34 and 35 can also be found using the method of characteristics, which assumes that the dependent variables S_c and C_{aq} can be derived along a path or characteristic on which one of the independent variables is a function of the other independent variable. In this case, we assume that r is implicitly a function of time, such that $S_c(r,t)$ and $C_{aq}(r,t)$ become $S_c(r_{sc}(t),t,r_0,t_0)$ and $C_{aq}(r_{aq}(t),t,r_0,t_0)$, where $r_{sc}(t_0) = r_{aq}(t_0) = r_0$ (for further details on this method of solution we refer the reader to (Haberman, 2004)). By making this assumption, the total derivatives for S_c and C_{aq} can be written as

$$\frac{dS_c}{dt} = \frac{\partial S_c}{\partial t} + v_{sc}\frac{\partial S_c}{\partial r} = \frac{\partial S_c}{\partial t} + \frac{dr_{sc}}{dt}\frac{\partial S_c}{\partial r_{sc}} \qquad \textbf{Eq. (36)}$$

and

$$\frac{dC_{aq}}{dt} = \frac{\partial C_{aq}}{\partial t} + v_{aq}\frac{\partial C_{aq}}{\partial r} = \frac{\partial C_{aq}}{\partial t} + \frac{dr_{aq}}{dt}\frac{\partial C_{aq}}{\partial r_{aq}}, \qquad \textbf{Eq. (37)}$$

where v_{sc} and v_{aq} are the characteristic wave speeds of each function [$m \cdot s^{-1}$]. Comparing these total derivatives to equations 34 and 35, we now find that the two

coupled partial differential equations in 34 and 35 can be written as the following set of four coupled ordinary differential equations

$$\frac{dS_c}{dt} = \frac{-h(1-S_c)\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c}\left(\frac{1}{1+\eta f} - 1\right), \qquad \textbf{Eq. (38)}$$

$$\frac{dr_{sc}}{dt} = v_{sc} = \frac{1}{\phi}\left(\frac{Q_{tot}}{2\pi H r_{sc}} + \frac{1}{r_{sc}}\int_{r_o}^{r_{sc}} -r' D dr'\right)\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right], \qquad \textbf{Eq. (39)}$$

$$\frac{dC_{aq}}{dt} = h\left(C_{aq}^{+} - C_{aq}\right), \qquad \textbf{Eq. (40)}$$

and

$$\frac{dr_{aq}}{dt} = v_{aq} = \frac{u_{db}}{\phi(1-S_c)}. \qquad \textbf{Eq. (41)}$$

For simple wave equations this separation method produces analytic solutions; however, here, this transformation does not lead to a closed form analytical solution for two reasons. First, the characteristic wave speeds of the functions S_c and C_{aq} are directly coupled, meaning that v_{sc} and v_{aq} both require information about S_c and C_{aq}. Second, at any position and time, solving $v_{sc}\left(r_{sc}(t)\right)$ requires an integration of all of the dissolution which has occurred between r_o and r_{sc}. As a result, although equations 38 - 41 describe the spatial and temporal evolution of both the SCFP CO_2 and the aqueous phase CO_2 along the paths $r_{sc}(t)$ and $r_{aq}(t)$, they must be solved numerically (code provided in Appendix 5). However, as we will demonstrate, describing the dynamics by this method will allow us to gain analytical insights from the wave speed and shock

speeds as they relate to the conditions necessary for brine imbibition to be effective at trapping SCFP CO_2 within the reservoir in a manner not possible using only TOUGH2.

The method of numerical solution to equations 38 - 41 is similar to Lagrangian based particle tracking. Instead of hard bodied particles, here we track discrete fluid packets representing kinematic waves of either the saturation of SCFP CO_2, S_c, or the mass concentration of dissolved CO_2, C_{aq}. These fluid packets are continuously added into the simulation at the injection well of the reservoir, where they receive a function value, $S_c(r_o, t)$ and $C_{aq}(r_o, t)$, consistent with the injection schedule. After being added to the simulation, both types of packets are tracked as they leave the injection well and convect outward into the reservoir. In brief, at each time step in the simulation, each S_c packet advances in the r direction with a velocity set by its characteristic velocity (equation 39). Over the same time step, the value of S_c stored within that packet changes according to equation 38. This procedure of updating the position and value during a single time step is repeated for all individual S_c fluid elements. Packets that track C_{aq} within the reservoir are updated in the same manner using equations 41 and 40, respectively. As mentioned above, updating S_c and C_{aq} requires information from both functions; however, since each fluid packet contains only information about one of these functions, when updating S_c packets, C_{aq} values are interpolated from nearest neighbor C_{aq} packets, and vice-versa.

In some cases, the solution procedure outlined above will be sufficient to model the evolution of SCFP and aqueous CO_2. However, as is often the case in wave equations solved with the method of characteristics, the above method alone is

insufficient in the event that two fluid particles of the same type (S_c or C_{aq}) attempt to occupy the same position in space at the same time. In general, when two or more characteristic waves intersect the function becomes multivalued and discontinuous. These intersections in the solution are called shocks and although they are formed at a particular point in space and time, once they are formed, these discontinuities will persist within the domain and will convect with their own velocity. For example, the white dashed line in Figure 4a(iii) represents a shock between S_c^- near the injection well and S_c^+ ahead of the brine flooding zone. Here, the shock velocity is found using the conservation of mass on a moving control volume that brackets the shock. In the context of the numerical solution procedure outlined above, when a shock is formed within the reservoir, the fluid packets which collided to form the shock are removed from the simulation and replaced with a single shock packet. Once introduced into the simulation, the velocity of a shock packet formed by the collision of two S_c packets is given by

$$w_{sc} = \frac{1}{\phi}\frac{(u_{dc})_- - (u_{dc})_+}{(S_c)_- - (S_c)_+}, \qquad \textbf{Eq. (42)}$$

where the subscripts (-) and (+) refer to the values of u_{dc} and S_c behind and in front of the shock, respectively. Equation 42 is derived from the conservation of mass of SCFP CO_2 around the shock wave. Similarly, the collision of two C_{aq} packets results in a shock wave whose velocity, obtained from conservation of dissolved CO_2 mass, is given by

$$w_{Caq} = \frac{1}{\phi}\frac{(C_{aq}u_{db})_- - (C_{aq}u_{db})_+}{(C_{aq}S_b)_- - (C_{aq}S_b)_+}. \qquad \textbf{Eq. (43)}$$

As these shock waves are tracked within the simulation, any S_c or C_{aq} packets which collide with a shock wave are removed from the simulation. Equations 24 - 27 and 38 - 43 now constitute a closed set of equations to describe the evolution of CO_2 within the reservoir during the initial injection process and during subsequent brine flooding.

2.2.4 Injection modeling using finite volume methods

The analytic approach leading to equations 38 - 43 captures the physical processes most important to modeling our proposed injection strategy: residual trapping occurring during the imbibition of brine, and local dissolution of CO_2 into the surrounding brine. However, it is important to note that these equations assume that the fluids are incompressible and nonreactive with the porous matrix, gravity is neglected, and the reservoir is isothermal, isotropic, and homogeneous. In order to relax the assumptions of incompressibility, isothermal reservoir conditions, and neglected gravity, we also used the commercial software package PetraSIM to model the injection strategies described in Figure 2. PetraSIM provides a GUI to create and mesh a geologic reservoir and then interfaces with TOUGH2-ECO2N to solve the coupled mass-transport and thermodynamic equations of states for the CO_2 and brine within the domain. TOUGH2, which uses integral finite difference formulations to solve these equations simultaneously under the assumption that mass phases reach thermodynamic equilibrium instantaneously in every differential control volume, has been used extensively in the literature to model various CO_2 dynamics within geologic reservoirs (Doughty, 2010, 2007; Doughty et al., 2009; Jordan and Doughty, 2009; Rodosta et al.,

2014; Spycher and Pruess, 2009; Xu et al., 2010). Using built-in equation of state models, TOUGH2 solves equations 14 - 16 directly, subject to the boundary conditions and injection schedule set by the user, including the effects of compressibility, gravity, and temperature dependent fluid properties such as solubility and viscosity. Table 3 summarizes the mesh and model inputs used in our TOUGH2 analysis.

initial brine pressure within the reservoir, P_o [Pa]	20 x 10^6	permeability, κ [m^2]	1 x 10^{-14}
reservoir size (radially symmetric) [m]	10 x 1500 (H x R)	porosity, ϕ [-]	0.15
grid size [m]	0.5 x 6 (H x R)	initial temperature of reservoir and all injected fluids [°C]	45
maximum CO_2 saturation, S_c^+ [-]	0.80	minimum CO_2 saturation, S_c^- [-]	0.25
injection cell mass flow rate into reservoir [$kg \cdot s^{-1}$]	0.3	far-field boundary condition (at 1500 m), fixed pressure [Pa]	20 x 10^6

Table 3. PetraSIM-TOUGH2 input parameters used for all simulations

2.2.5 Description of the CO_2-brine injection schedule

To investigate the effects of brine flooding, several injection schedules were modeled using the analytic equations and methodology discussed in sections 2.2.3 and 2.2.4. These injection schedules differed in the ratio of injected brine volume:CO_2 volume, γ, which was varied from 0 to 21.25. The ratio $\gamma = 0$ represents the base case: a conventional injection strategy represented in Figure 2a. Each schedule began with a ~23,300 metric tonne CO_2 injection (3 $kgCO_2 \cdot s^{-1}$ for 90 days). The total mass and flow rate is representative of a small, pilot-scale operation. Following this initial CO_2

injection, brine was injected into the reservoir at a flow rate of 3 kg·s^{-1} until the prescribed volume ratio was achieved. All injection schedules were modeled twice for different geologic conditions, once using Corey exponents $n = m = 1$ and again using $n = m = 4$, as most Corey exponents published in literature fall between these values (Bachu, 2013; Burnside and Naylor, 2014).

At the end of each injection schedule, we report the mobile, trapped, and dissolved mass fractions of CO_2 within the entire reservoir. These mass fractions will be defined as

$$X_{mobile} = \frac{\phi}{\dot{m}_{CO2} * t_{inj}} \int \rho_c(r,z) * \max\{S_c(r,z) - S_c^-, 0\}\ dV\ , \qquad \textbf{Eq. (44)}$$

$$X_{trapped} = 1 - X_{mobile} - X_{dissolved}\ , \qquad \textbf{Eq. (45)}$$

and

$$X_{dissolved} = \frac{\phi}{\dot{m}_{CO2} * t_{inj}} \int C_{aq}(r,z) * \left(1 - S_c(r,z)\right) dV, \qquad \textbf{Eq. (46)}$$

where $\dot{m}_{CO2}$ is the CO_2 mass flow rate during the initial CO_2 injection period [kg·s^{-1}] and t_{inj} is the total CO_2 injection time [s]. Here, we define the mobile fraction of CO_2 within a given region as the fraction of CO_2 above S_c^-, since it is assumed that if the CO_2 were to migrate it would leave behind pockets of residually trapped gas at a volume set by S_c^- (Burnside and Naylor, 2014). The logic operator, *max*, is required in the integrand of equation 44 because in regions where the local CO_2 saturation has fallen below the minimum residual saturation, for example due to dissolution, the argument $S_c(r,z) - S_c^-$ becomes negative. We also note that in applying equations 44 and 46 to

the numerical model described by the method of characteristics, ρ_c is a constant and the functions S_c and C_{aq} are only functions of r, since the equations were derived for a radially symmetric reservoir.

2.3. Discussion and results

2.3.1 Predictions from the method of characteristics

Before solving equations 38 - 43 numerically with the procedures outlined in section 2.2.3, these equations provide valuable insights into the dynamics of brine flooding and allow us to predict how effectively brine injection can immobilize a plume of CO_2. In this context, the effectiveness of brine flooding can be defined as the ability of the brine to trap SCFP CO_2 into residual pockets. If the brine were 100% effective, then the result would be a reservoir containing CO_2 at S_c^- everywhere in the domain. Flooding will therefore be less effective if mobile CO_2 ($S_c > S_c^-$) remains in the reservoir after brine has been injected.

We can estimate the effectiveness of brine flooding in a very simple manner by recognizing the relationship between the characteristic wave speed, v_{sc}, and the shock speed, w_{sc}. First, let us ignore dissolution. Omitting dissolution from this estimate results in a worst case scenario, since any dissolution would aide in reducing the amount of mobile SCFP CO_2 in the reservoir. Without dissolution, the only means of immobilization is the mechanism of trapping the SCFP CO_2 in residual pockets.

Neglecting dissolution is also convenient mathematically, as it implies $h = D = 0$ and equations 38 - 43 strictly reduce to

$$\frac{dS_c}{dt} = 0 \qquad \textbf{Eq. (47)}$$

$$v_{sc} = \frac{1}{\phi}\left(\frac{Q_{tot}}{2\pi H r_{sc}}\right)\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right] \qquad \textbf{Eq. (48)}$$

$$w_{sc} = \frac{1}{\phi}\frac{(u_{dc})_- - (u_{dc})_+}{(S_c)_- - (S_c)_+}. \qquad \textbf{Eq. (49)}$$

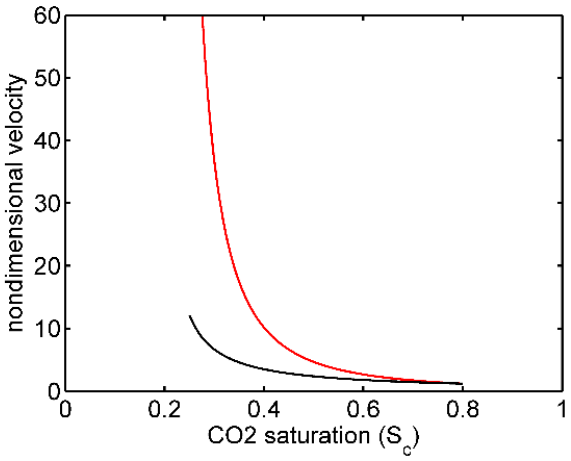
If we assume that the CO_2 initially injected into the reservoir efficiently displaces the native brine, then the CO_2 saturation within the plume near the well will be approximately S_c^+ (Figure 4a(ii)). When CO_2 injection ends and brine injection begins, the majority of the CO_2 adjacent to the well will be displaced further into the reservoir but some of the CO_2 will remain in isolated, randomly dispersed pockets, represented in Figure 4a(iii). The interface, depicted as the white dashed line in Figure 4a(iii), between this low saturation region, on the left, and this high saturation region, on the right, represents a discontinuity, or shock, in the function $S_c(r,t)$. The rate at which this interface advances into the reservoir is given by equation 49 and is set by the CO_2 saturation in front of and behind the shock. In front of the interface we have assumed that $S_c = S_c^+$. Ideally, we would like the CO_2 saturation behind this interface to be S_c^-, as this would correspond to the complete immobilization of CO_2 within the brine flooded zone. However, in order for a shock to persist between fluid regions of S_c^+ in front and S_c^- behind, fluid regions of S_c^- must have a characteristic velocity greater than or equal to the velocity of the shock, expressed as $v_{sc}(S_c^-) \geq w_{sc}(S_c^-, S_c^+)$. If this

condition is not satisfied, then the saturation behind the shock cannot remain at S_c^-. Instead, the CO_2 saturation behind the shock must increase until $v_{sc}(S_c) = w_{sc}(S_c, S_c^+)$. Returning now to the question of brine flooding effectiveness, using equations 48 and 49, the nondimensional characteristic velocity, $\tilde{v}_{sc}(S_c)$, and nondimensional shock velocity, $\tilde{w}_{sc}(S_c, S_c^+)$, have been plotted for four sets of Corey exponents, $n = m = \{1,1.2,2,4\}$, in Figure 5, where $\tilde{v}_{sc} = v_{sc}/(Q_{tot}/2\pi Hr\phi)$ and $\tilde{w}_{sc} = w_{sc}/(Q_{tot}/2\pi Hr\phi)$, respectively. For the cases of $n = m = 1$ and $n = m = 1.2$, it is evident that $v_{sc}(S_c^-) \geq w_{sc}(S_c^-, S_c^+)$, indicating that brine flooding is predicted to completely immobilize CO_2. However, in the cases of $n = m = 2$ and $n = m = 4$, $v_{sc}(S_c^-) < w_{sc}(S_c^-, S_c^+)$. In these cases brine flooding cannot be expected to fully trap the CO_2. Instead, the CO_2 saturation behind the shock should evolve towards $S_c \sim 0.28$ and $S_c \sim 0.37$, respectively, corresponding to the saturation at which $v_{sc}(S_c) = w_{sc}(S_c, S_c^+)$. Using these values of CO_2 saturation, the fraction of mobile CO_2 remaining behind the shock can be estimated by $(S_c - S_c^-)/S_c$, which for these two cases are 9.3% and 32.2%, respectively. This analysis was repeated for a range of CO_2 Corey exponents from 1 to 7, consistent with exponent values found in experiments (Bachu, 2013). As shown in Figure 6, there is a clear trend that larger n values are expected to result in more mobile CO_2 remaining after brine flooding. In the extreme case of $n = 7$, brine flooding is predicted to only immobilize 50% of the CO_2. Figure 6 also depicts the importance of the brine Corey exponent, m. As m becomes smaller, we predict larger fractions of mobile CO_2 following brine flooding, although these effects are less pronounced.

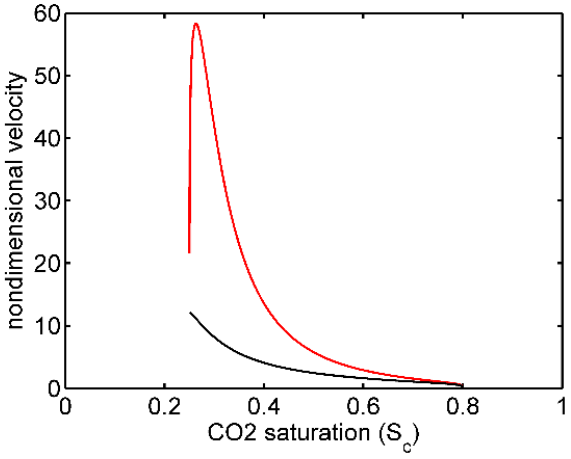
The dependence of brine effectiveness on Corey exponent can be explained with this simplified model. From equation 48, the Corey exponents effect the shape of the

characteristic velocity functions through the derivative of $1/(1+\eta f)$ with respect to CO_2 saturation. When the Corey exponents are larger (greater curvature in the relative permeability functions, Figure 4b), the function $\partial(1/(1+\eta f))/\partial S_c$ becomes small and lowers the characteristic velocity of the SCFP CO_2. This analysis provides a mechanistic connection between the microstructure of the formation and injection dynamics, and would be difficult to extract from the fully numerical approaches of standard reservoir simulators.

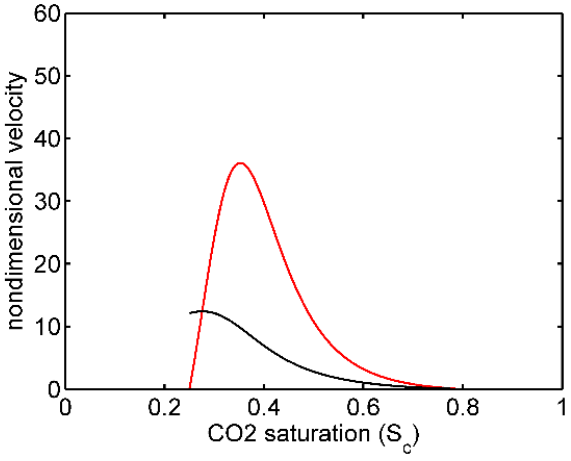
a)


nondimensional velocity
CO2 saturation (S_c)
0
10
20
30
40
50
60
0
0.2
0.4
0.6
0.8
1

b)


nondimensional velocity
CO2 saturation (S_c)
0
10
20
30
40
50
60
0
0.2
0.4
0.6
0.8
1

c)


nondimensional velocity
CO2 saturation (S_c)
0
10
20
30
40
50
60
0
0.2
0.4
0.6
0.8
1

d)

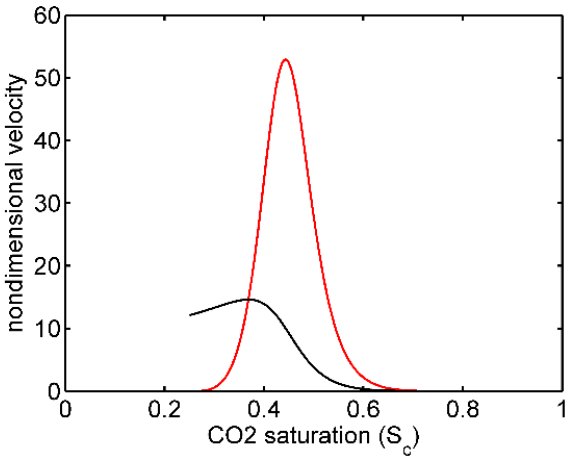


Figure 5. The nondimensional characteristic velocity of SCFP CO_2, $\tilde{v}_{Sc}$, shown in red, and nondimensional shock velocity, $\tilde{w}_{Sc}$, shown in black, in the absence of dissolution, using the Corey exponents (n and m) (a) 1, (b) 1.2, (c) 2, and (d) 4.

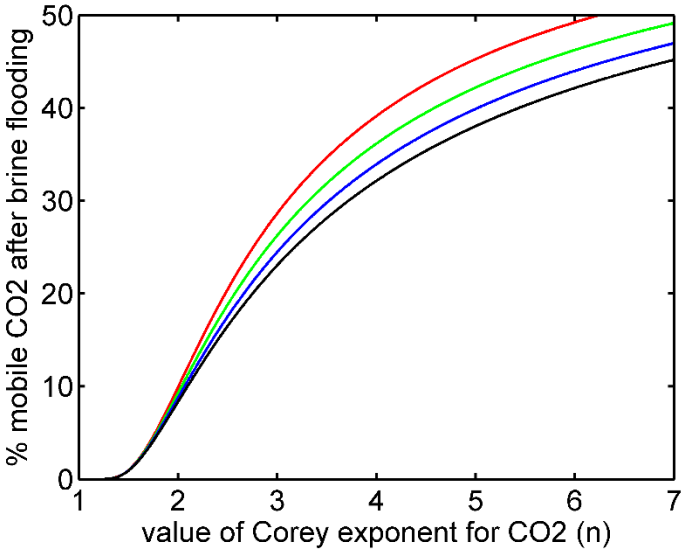


Figure 6. The estimated mobile fraction of SCFP CO_2 following brine flooding, in the absence of dissolution, as a function of the CO_2 Corey exponent, n, using the brine Corey exponent, m, of 1 (red), 2 (green), 3 (blue), and 4 (black). The mobile fraction was estimated by finding the SCFP CO_2 saturation required to satisfy $v_{sc}(S_c) \geq w_{sc}(S_c, S_c^+)$, as discussed in section 2.3.1.

2.3.2 Results from modeling the dynamics of brine flooding

To analyze the dynamic effects of brine flooding, the numerical modeling methods discussed in sections 2.2.3 and 2.2.4 were used to simulate CO_2 being injected into a geologic reservoir at a flow rate of 3 $kg \cdot s^{-1}$ for 90 continuous days. The S_c and C_{aq} distribution within the reservoir at the end of this 90 days represents the base case, $\gamma = 0$, which corresponds to a conventional sequestration process, represented in Figure 2a. Following this initial injection of CO_2, brine was injected at a flow rate of 3 $kg \cdot s^{-1}$ for 2,272 continuous days, which corresponds to a final γ value of 21.25. Throughout the brine injection, the mass fractions of mobile, trapped, and dissolved CO_2 were calculated according to equations 44 - 46.

Figure 7 presents the spatial distributions of S_c and C_{aq} for selected values of γ during brine injection, using the Corey exponents $n = m = 1$ (left column) and $n = m = 4$ (right column). In this figure, the red and black lines correspond to the method of characteristics solution, while the blue lines correspond to the TOUGH2 results which have been averaged, locally in the z-direction. The results in Figure 7a correspond to $\gamma = 0$ (no brine injection). Without any brine injection, we find that the majority of SCFP CO_2 is mobile with local CO_2 saturation values well above S_c^- (denoted by the lower dotted line in the SCFP plots). As brine is injected into the reservoir there is a rapid decrease in the fraction of mobile CO_2 as the plume is dispersed into the reservoir and residually trapped (Figure 7b). The majority of CO_2 trapping is finished before reaching a brine:CO_2 volume ratio of 2.75 (Figure 7c). For $\gamma > 2.75$, the plume continues to dissolve until all of the sequestered CO_2 is in the aqueous phase (Figure 7d-f).

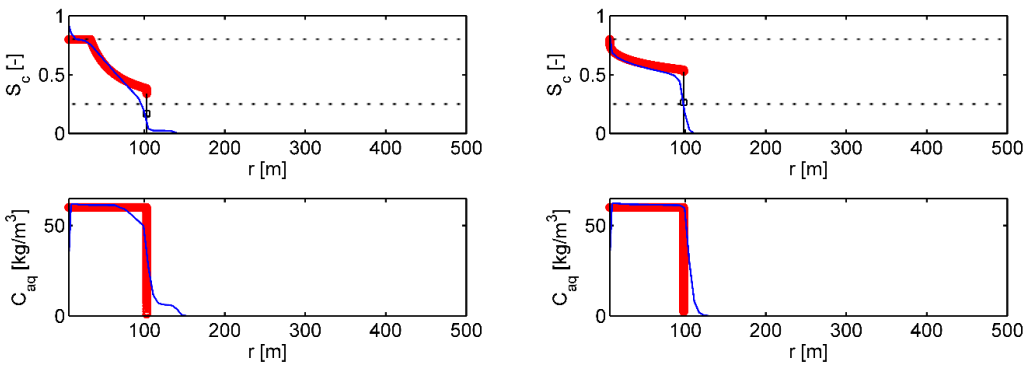
Figure 7 also highlights two more subtle results. First, there are differences in the dynamics between the Corey exponents 1 and 4 relating back to their differences in characteristic velocities and shock velocities. As brine is first injected (starting from Figure 7a and transitioning to Figure 7b), there is a shock within the SCFP CO_2 domain that originates from the injection well and travels outward through the CO_2 plume with a velocity given by equation 42. Behind this shock, the local saturation of CO_2 is greatly reduced. For the case of Corey exponents $n = m = 1$ (left column), this shock is present in Figure 7b at approximately $r = 100\ m$. In this case, the saturation behind this shock is reduced to S_c^-. As brine injection continues and $\gamma = 1$ (Figure 7c), this shock has overtaken the front of the CO_2 plume and all remaining SCFP CO_2 has been immobilized via residual trapping. This result is shown both by the method of characteristics solution and by the TOUGH2 model. In the case of Corey exponent $n = m = 4$ (right column), the shock created when brine is first injected into the reservoir has a much larger velocity and, as a result, has already swept through the reservoir before $\gamma = 0.5$ (Figure 7b). More importantly, however, is the state of the SCFP CO_2 behind this shock. As predicted and discussed in section 2.3.1, brine flooding is unable to completely trap the SCFP CO_2 residually, as depicted in Figure 7b. In fact, a small fraction of mobile SCFP CO_2 remains in the reservoir beyond $\gamma = 5$ (Figure 7e),; a result consistent with the predictions of the trend in effectiveness with Corey exponent depicted in Figure 6. Second, although the TOUGH2 results and the method of characteristic results are in close agreement to one another, the inclusion of gravity in the TOUGH2 model allows us to quantify the effects of vertical CO_2 migration within the reservoir. Rising CO_2 maintains higher SCFP CO_2 saturation values near the top of

the reservoir and as a result, this fraction of CO_2 has greater mobility and convects faster in the radial direction; the effect of this increased mobility is seen in the leading edge structure that emerges in the TOUGH2 prediction (blue curve in frames a-e). As a consequence, the CO_2 plume will occupy a greater reservoir footprint than the areas predicted by equations 32 and 33. However, these effects are less pronounced for the case of Corey exponents $n = m = 4$.

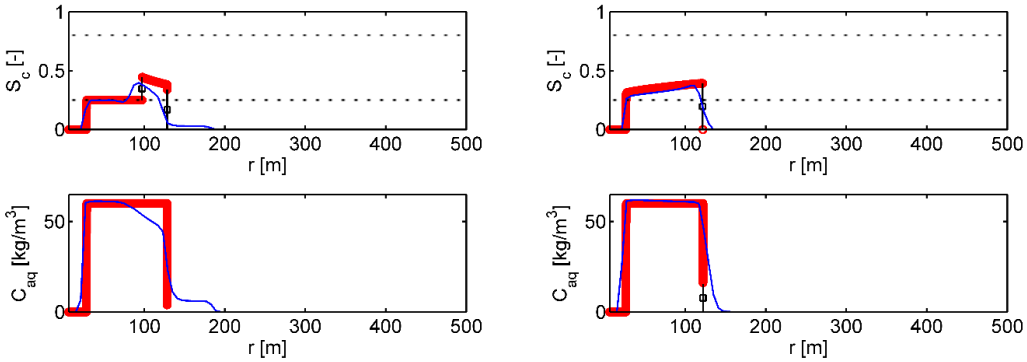
This CO_2 pooling near the top of the reservoir is best observed by considering the complete two dimensional results from the TOUGH2 model, presented in Figure 8. Here, it is apparent that a very thin region of mobile CO_2 exists directly beneath the caprock, particularly in the case of Corey exponents $n = m = 1$.

Corey exponents $n = m = 1$ | Corey exponents $n = m = 4$

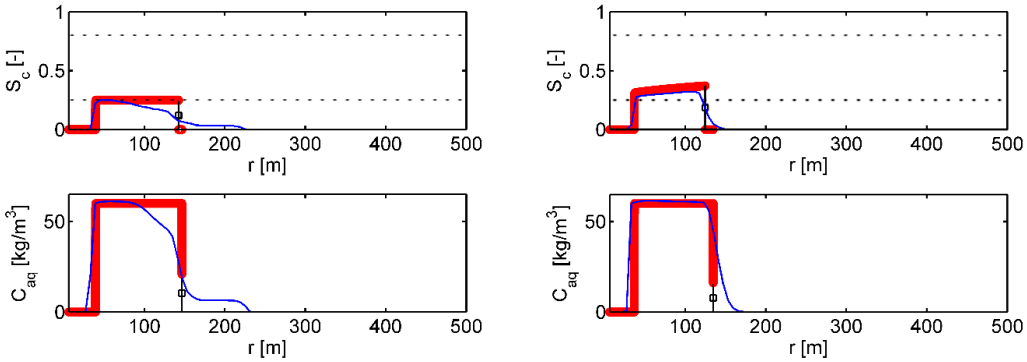
a)



b)



c)



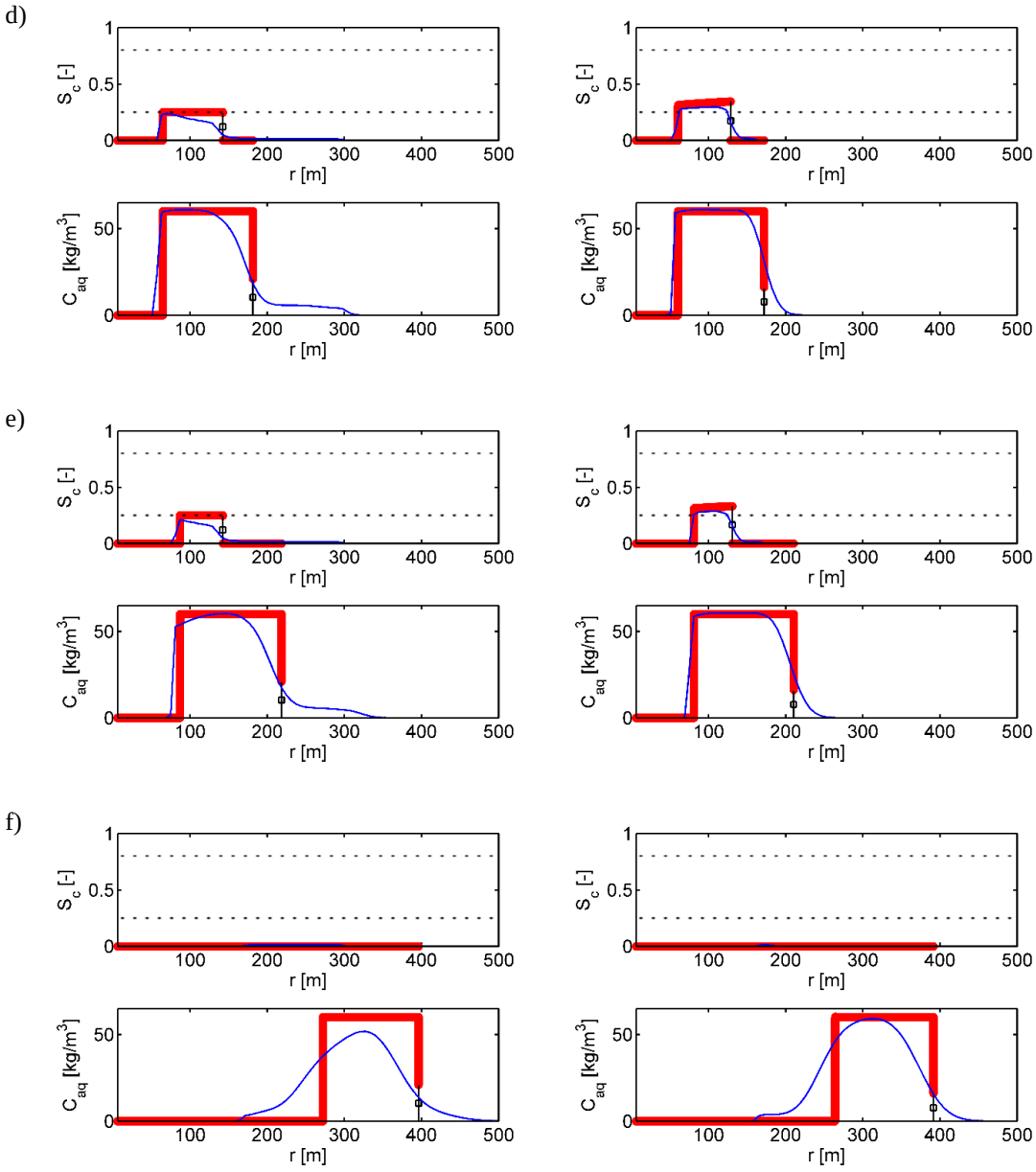


Figure 7. The spatial distribution of S_c and C_{aq} within the reservoir during the injection process from the method of characteristics solution (red) and the z-averaged TOUGH2 solution (blue) at (a) $\gamma = 0$, corresponding to the end of CO_2 injection, (b) $\gamma = 0.5$, (c) $\gamma = 1$, (d) $\gamma = 2.75$, (e) $\gamma = 5$, and (f) $\gamma = 21.25$. The results shown are using the Corey exponents $n = m = 1$ (left column) and $n = m = 4$ (right column). In all plots, solid black vertical lines denote a shock in the method of characteristics solution and the square marker corresponds to the average value (either of S_c or C_{aq}) between the front and the back of the shock wave. The dashed horizontal lines in the S_c plots correspond to the values S_c^- and S_c^+.

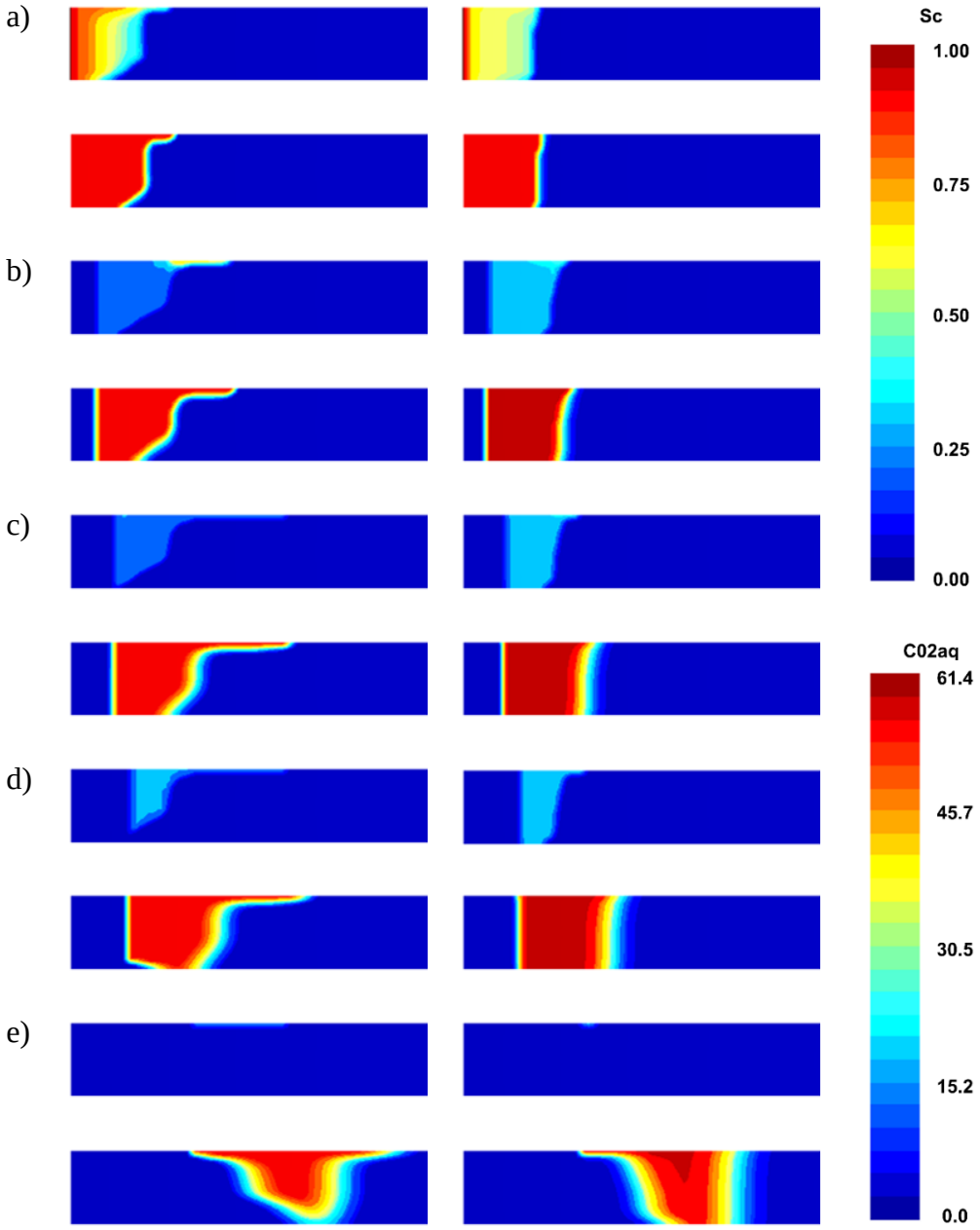


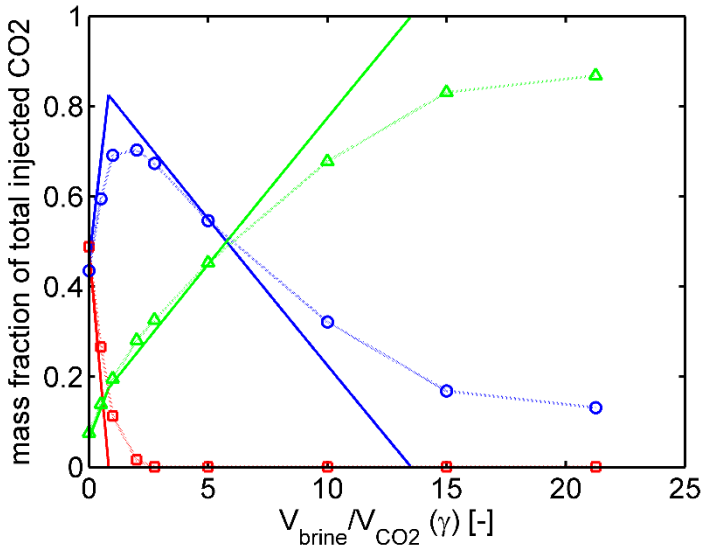
Figure 8. Spatial distribution of S_c (top images in each series) and C_{aq} (bottom images in each series) on the domain $0 \leq r \leq 500$ (horizontal direction) and $0 \leq z \leq H$ (vertical direction) within the reservoir during the injection process at (a) $\gamma = 0$, (b) $\gamma = 1$, (c) $\gamma = 2.75$, (d) $\gamma = 5$, and (e) $\gamma = 21.25$ for Corey exponents $n = m = 1$ (left column) and Corey exponents $n = m = 4$ (right column). These results come from the simulations performed in TOUGH2, described in section 2.2.4.

Finally, the mobile SCFP, trapped SCFP, and dissolved aqueous mass fractions of CO_2 were calculated as a function of γ according to equations 44 – 46 in both the numerical simulations based on the method of characteristics and TOUGH2. These results are presented in Figure 9. Overall, the method of characteristics solution is qualitatively similar across the entire range of parameters to the results from TOUGH2 and quantitatively similar for low γ for $n = m = 1$ and for all γ for $n = m = 4$. However, in the case of $n = m = 1$, these two methods begin to differ significantly for $\gamma > 12$. This can be explained by the buoyancy driven pooling evident in Figure 8. Because some of the CO_2 collects below the caprock, most of the injected brine convects below this remaining plume and therefore does not aid in dissolution. Since gravity is ignored in the method of characteristics solution, the CO_2 is evenly distributed in the z direction and all injected brine sweeps efficiently through the entire plume. As a result, regardless of the Corey exponents, the method of characteristics solution contains zero SCFP CO_2 for $\gamma > 14$, whereas the TOUGH2 simulations show a small fraction of immobile SCFP CO_2 above $\gamma = 14$, with this mass fraction being larger in the case of $n = m = 1$, where greater CO_2 vertical migration has occurred. Including compressibility in the TOUGH2 model had a negligible effect on the dynamics and outcome of the injection process compared to the method of characteristics. The density of the CO_2 increased near the injection well as a result of increases in injection pressure, however, without changing the interfacial surface tensions between brine and CO_2, the dynamics of trapping residual CO_2 remains unchanged regardless of the CO_2 density.

Most significantly, brine flooding in this manner is able to reduce the mass fraction of mobile SCFP CO_2 to less than 10% using $\gamma \leq 2.75$ in both sets of Corey

exponents considered. However, the values of the Corey exponents play a clear role in the dynamics of brine flooding. For $n = m = 1$, it is possible to completely immobilize the SCFP CO_2 plume using a volume ratio of only $\gamma \approx 2$. All brine injection beyond this volume ratio will simply aide in dissolving the trapped CO_2. For $n = m = 4$, some of the SCFP CO_2 remains mobile throughout the entire brine injection process, almost until it is completely dissolved. These dynamic results are consistent with the discussion in section 2.3.1 regarding characteristic and shock wave speeds. Namely, at larger values of n and m, the shock wave leading the brine flooded zone between the low CO_2 saturation zone, behind, and the highly CO_2 saturated zone, ahead, cannot be maintained with a value of S_c^-. The CO_2 saturation behind this shock must inevitably increase until $v_{sc} = w_{sc}$.

a)



b)

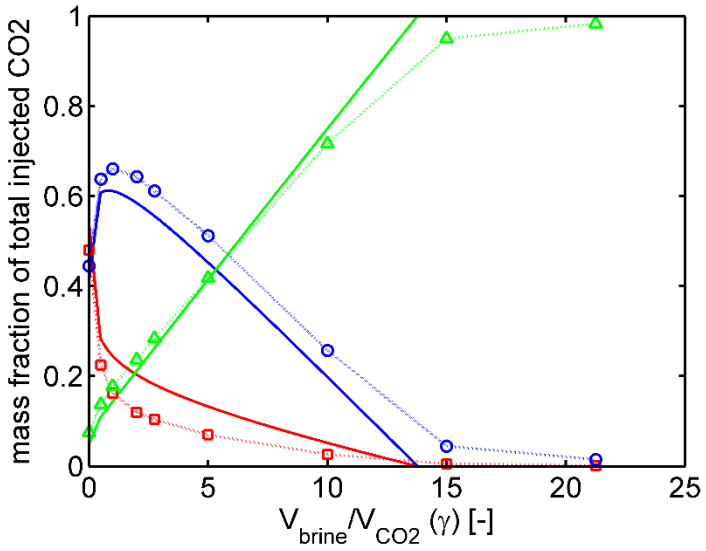


Figure 9. Mobile SCFP, trapped SCFP, and dissolved aqueous mass fractions of CO_2 (equations 44 - 46), shown in red, blue, and green, respectively, within the reservoir as a function of γ for (a) Corey exponents $n = m = 1$ and (b) Corey exponents $n = m = 4$. In both plots, the dark, solid lines correspond to the results using the method of characteristics, while the data points correspond to individual simulation results from our analysis using TOUGH2. The lighter, dashed lines linearly connect the TOUGH2 data points.

2.4. Conclusion

In addition to the full dynamic simulations presented here, the analytic tools described in sections 2.2.2 and 2.3.1 provide simple methods to estimate brine volume requirements and predict the effectiveness of brine flooding a priori simply by knowing a few key properties of the porous media such as S_c^-, S_c^+, n, and m. These methods clarify the importance of shock dynamics in controlling the process of brine flooding and allows for the dissection of the different effects when compared to TOUGH2.

The safe storage of carbon dioxide in the geologic subsurface remains a challenging proposition. Long after a conventional injection period ends, plume mobility can persist and increase the risk of leakage from the reservoir. Brine flooding presents one possibility to mitigate plume mobility by rapidly inducing residual trapping and could also be used to fully dissolve the CO_2 plume. The benefits of brine flooding are dependent on many reservoir characteristics including the relative permeability of CO_2 and brine within the porous media. Here, we have shown that injecting brine volumes of 2.75 times that of the CO_2 volume reduces the mass fraction of mobile SCFP CO_2 to less than 10% in a time span that is only 5 times longer than the original CO_2 pumping time. This rate of decay is significantly faster than the time required for this mobility loss to occur naturally. Furthermore, the majority of the injected CO_2 can be completely dissolved using brine volume ratios of 14 – 20, depending on the severity of vertical CO_2 migration within the reservoir. However, achieving these results requires the additional costs associated with the harvesting and reinjection of brine.

In the context of reservoir engineering, selecting an ideal reservoir to sequester and brine-flood CO_2 will depend upon the desired state of the CO_2 plume at the end of

brine injection. From the predictions presented here, trapping SCFP CO_2 using brine flooding is most effective when the relative permeability of CO_2 and brine are characterized by small Corey exponent. However, in these cases, the CO_2 plume will rise more readily and make complete dissolution difficult. Conversely, if the goal of brine flooding is to completely dissolve the plume, one should look for reservoirs in which the relative permeability of CO_2 and brine are characterized by large Corey exponent. Even though such conditions exclude the possibility of complete residual trapping, they reduce vertical migration and enable effective dissolution to occur. Alternatively, the effectiveness of brine flooding would increase by sequestering CO_2 at higher densities such that the density difference between the CO_2 and native brine is small, thereby minimizing buoyancy driven vertical migration. Of course, these performance predictions require knowledge of the brine and CO_2 relative permeability throughout the reservoir and will undoubtedly be a point of uncertainty in modeling flooding dynamics due to the difficulty and expense of such measurements.

We would like to conclude with a brief discussion of the parameters used in this study. While we have chosen to let $\phi = 0.15$ and $\kappa = 1 \times 10^{-14}\, m^2$ throughout our analysis, actual values of rock porosity and permeability vary widely; and although the brine flooding process described by the solution to the method of characteristics is broadly independent of these parameters, rocks described by extremely low porosity may also exhibit relative permeabilities not well described by the Brooks-Corey model presented here. Furthermore, smaller values of absolute permeability tend to restrict injectivity of all fluids, and consequently require higher injection pressures to achieve a prescribed flow rate. At the same time, low permeability causes greater pressures to be

required to inject the working fluid, which would lead to greater compressibility effects in the CO_2 phase and may also increase the risk of reservoir fracturing near the injection well. For reference, our TOUGH2 simulations indicated that ΔP_{pump} became as high as 18 MPa during brine injection for the case of Corey exponent 4, while ΔP_{pump} became as high as 13 MPa during CO_2 injection alone. This value of 18 MPa is roughly four times larger than that of the simple estimate of ΔP_{pump} described in section 2.2.1. While this value is generally less than the fracturing stress of most reservoirs, the acceptable threshold will vary depending on the rock formation and the relationship between κ and ΔP_{pump} should be noted.

Finally, we note that in this work, the value of the mass transfer capacity coefficient, h, used in the method of characteristics analysis was purposefully selected to achieve fast dissolution relative to the global convection of SCFP CO_2 and brine. This was done for two reasons: (1) most authors who model CO_2 sequestration treat dissolution as a process which happens instantaneously over small length scales, and (2) it allowed us to use our TOUGH2 analysis as a way to elucidate compressibility and gravity effects while ignoring dissolution kinetics, since TOUGH2 explicitly assume that dissolution is instantaneous within each grid cell. However, equations 38 - 43 are written in a general form, and could be used to model the dynamics of any system, including those with slow dissolution kinetics.

CHAPTER 3

MODELING THE DYNAMICS OF REMOBILIZED CO_2 WITHIN THE GEOLOGIC SUBSURFACE

3.1. Introduction

The total volume of carbon dioxide injected into the subsurface continues to grow as the US and other countries build and operate large-scale sequestration facilities. While these facilities may be considered as demonstration projects, further investment in sequestration technologies will likely lead to broader deployment in an effort to maintain fossil fuel use while simultaneously reducing anthropogenic atmospheric emissions. Sequestration may complement shifts toward renewable energy production, but its success, as a concept, relies upon sequestered CO2 remaining trapped in the subsurface for generations to come.

Because the concepts and process details of carbon sequestration have been well studied, we will not review them here; we direct the reader to several authors with published work on this topic (Huber et al., 2016; Huppert and Neufeld, 2014; Leonenko and Keith, 2008; Neufeld and Huppert, 2009; Pegler et al., 2014). However, for the purposes of this discussion, we summarize the following aspects of the sequestration process: (1) the majority of the storage capacity that is estimated to exist within the subsurface resides in brine aquifers at depths where the ambient temperature and pressure would maintain the CO2 in the supercritical phase at densities lower than that of the native brine, (2) during injection, the CO2 is mobile and it remains mobile long after injection stops, and (3) the degree to which CO2 is mobile and the duration of its

mobility is a complicated function of the reservoir geometry, the surface properties of the rock, the thermodynamic properties of the fluid phases involved, and the history of the CO2 plume motion (Huppert and Neufeld, 2014). This latter point has been the subject of numerous studies, where various authors have analyzed injection strategies using multiple injection wells and different well configurations deployed in various reservoir geometries (Akinnikawe et al., 2010; Anchliya and Ehlig-Economides, 2009; Hesse et al., 2008; Neufeld and Huppert, 2009; Shamshiri and Jafarpour, 2012). In general, these studies propose injection methods and identify optimal reservoir and well geometries aimed at decreasing the pressure build up within the formation or decreasing the time required for CO2 to become trapped by one of the four basic trapping mechanisms: structural, residual, solubility, and mineral. A full review of these mechanisms in the context of carbon sequestration has recently been discussed by (Huppert and Neufeld, 2014), but in brief: structural trapping requires a spatial variation in pore structure which results in an abrupt decrease in permeability and typically occurs at the boundaries of adjacent rock formations. Caprock seals located above a reservoir are the most prominent example of structural trapping, and are largely responsible for preventing injected CO_2 from immediately returning to the surface. Residual trapping occurs when isolated pockets of CO_2 are immobilized in pore spaces by capillary forces, and results from bulk CO_2 migrating within a porous medium in the presence of at least one other fluid phase. Finally, solubility and mineral trapping are possible depending on the specific properties of the reservoir and refer to the dissolution and chemical reaction of CO_2 with the native fluids and solid matrix, respectively.

For general injection scenarios, it has been predicted through numerical

simulations that in most reservoirs, mobile CO2 will slowly migrate through the formation due to buoyancy forces. As the plume migrates, the mobile fraction of CO2 decreases because it leaves behind residually trapped pockets of CO2 in the wake of its motion. Eventually the mobile volume becomes small and the buoyancy forces acting on the CO2 plume do not exceed the capillary forces resisting its motion at its boundaries, at which point the remaining plume is immobile (Golding et al., 2011; Hesse et al., 2008).

In this chapter, we consider a reservoir in which CO2 has been sequestered and allowed to become residually trapped through buoyancy-driven plume spreading, depicted in the schematic in Figure 10a and b(i). From this initial state, we suppose that an event occurs of natural origin (e.g. subsurface motion along an existing fault line) or man-made (e.g. producing fluid from an adjacent reservoir), and this event decreases the pore pressure in the CO2 reservoir. As a consequence, the local density of CO2 decreases causing the CO2 within the trapped pockets to expand. This expansion increases the saturation of CO2 to a value above the minimum residual saturation and, therefore, a portion of the CO2 becomes mobile. Due to density differences between the CO2 and native brine, this remobilized CO2 will migrate upward within the reservoir (Figure 10b (ii)) and pool beneath the caprock. As it collects at the roof of the reservoir it may also spread laterally (Figure 10b (iii)); this spreading could be problematic if there exist nearby pathways for the CO2 to escape the reservoir and continue migrating upward through the geologic strata. In this work, we therefore aim to address several important questions related to this scenario: (1) if the pressure within the reservoir decreases, what fraction of the residually trapped CO2 will become mobile, (2) how

much time is required for the mobile CO2 to rise and spread within the reservoir, (3) to what extent will the remobilized CO2 laterally spread into the far-field of the reservoir.

To answer these questions about remobilization, we require an accurate treatment of relative permeability as a function of saturation near the critical residual limit. To address this need, we propose and analyze a model of relative permeability that deviates from the commonly used Brooks-Corey model near the limit of low CO2 saturation in that it incorporates transport scaling laws based on percolation theory. Although the relative permeability in our model and the Brooks-Corey model are quantitatively very close in value, we demonstrate that the difference in scaling leads to significant differences in the predicted dynamics of remobilized CO2 for which saturation is close to the minimum residual value. In particular, using our model of relative permeability, we predict CO2 motion may occur 10^2 – 10^4 times faster than would be predicted using the more common Brooks-Corey model. To support the functional form and use of our model in this context, we present published experimental data of relative permeability of CO2 and brine from two different reservoir core samples and demonstrate that our model fits this data near the residual saturation limit better than the Brooks-Corey model that was originally used to fit the data.

To discuss and analyze the physics and dynamics of all steps in this remobilization process in appropriate detail, we have organized this chapter into three distinct sections. The first section addresses the estimated effect of remobilization from a depressurization event and provides scaling analyses of fluid migration time scales to support the decoupling of the vertical migration dynamics from the horizontal spreading dynamics. Within this section, we also present our proposed relative permeability

mobile to include percolation scaling near the residual saturation limit and we demonstrate that our proposed model more accurately describes experimental data compared with the Brooks-Corey model. The subsequent two sections explicitly derive the governing equations and numerical results related to our study of the vertical and horizontal migration of remobilized CO2, respectively, and compares how differences in rock properties, initial conditions, and choice of relative permeability function affect the predicted dynamics of the system and the steady state saturation conditions.

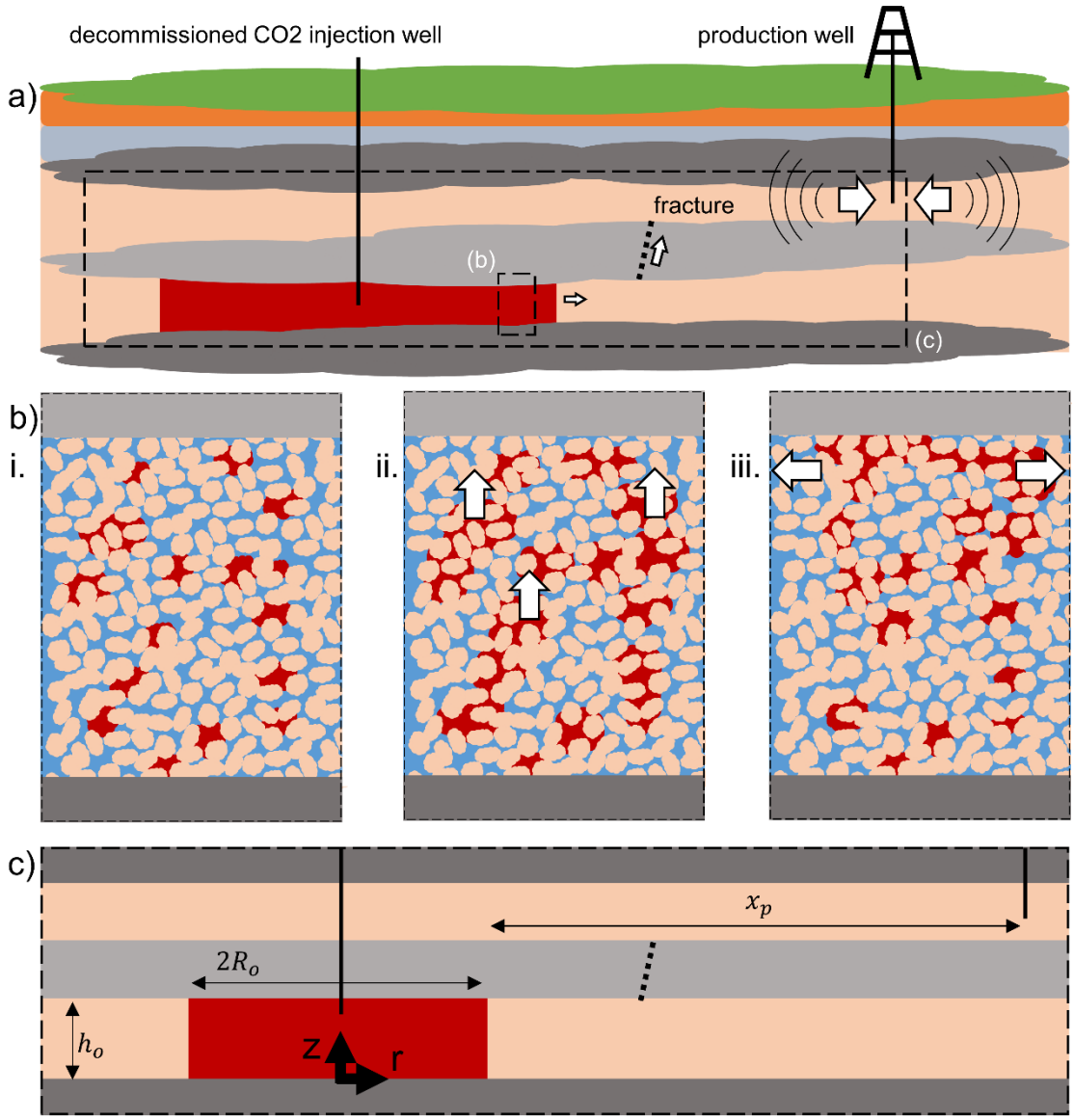


Figure 10. (a) At the reservoir scale, CO2 has been pumped into a formation and allowed to naturally migrate until the plume has become immobile. A depressurization event, such as the production of fluids (shown), causes a drop in pressure. Pressure information propagates to the CO2 through hydraulically connected pathways, such as a fault or fracture. (b) At the pore-scale, we assume (i) that the CO2 is initially trapped in residual pockets dispersed throughout the domain prior to the depressurization event. (ii) A decrease in pressure causes the trapped CO2 to increase in volume (decrease in density) and the local CO2 saturation increases such that a fraction of the CO2 is no longer trapped. This mobile CO2 will rise in the formation due to buoyancy and collect beneath the caprock. (iii) As the CO2 rises it will tend to spread along the caprock possibly leaking through fault lines. (c) Map of the axisymmetric geometry of the reservoir with geometrical parameters labelled.

3.2 Scaling analysis and modeling considerations of remobilized CO2

3.2.1 Estimating the mobile fraction of CO2 after a depressurization event

We begin our analysis by estimating the time scale for a pressure disturbance to propagate through the subsurface by poroelastice diffusion: $\tau_p \sim x_p^2/C$, where x_p [m] is the distance traveled by the pressure wave and C [$m^2 \cdot s^{-1}$] is the poroelastic diffusivity of the medium (given by $C = B\kappa/\mu\phi$). Using values representative of water in sandstone, $C = 0.15\ \mathrm{m}^2/\mathrm{s}$ ($B = 2.2$ GPa, $\kappa = 1 \times 10^{-14}\mathrm{m}^2$, $\phi = 0.15$, $\mu = 1 \times 10^{-3}\ \mathrm{Pa} \cdot \mathrm{s}$), and assuming a characteristic distance of 1 km, $\tau_p \sim 79$ days; we will later demonstrate that this pressure transient is much shorter than the time required for CO2 migration.

Next, we estimate the saturation of CO2 that would be present following a depressurization event. We assume prior to depressurization that the supercritical CO2 is immobile and trapped at the minimum residual saturation volume fraction, S_c^- [-], in disperse pockets distributed throughout the reservoir at an initial density, ρ_o [$kg \cdot m^{-3}$], set by the pressure and temperature of the domain, (Figure 10b (i)). From this initial state, we assume that pressure fluctuations will not greatly affect the total mass of supercritical CO2 present within a given region. In reality, neighboring brine will likely contain dissolved CO2 and the solubility of CO2 in brine does vary with pressure. However, we assume that over the range of pressures considered, the variation in solubility is small and, therefore, we will neglect mass transfer between CO2 dissolved in brine and the supercritical CO2. Consequently, a decrease in reservoir pressure will only affect the local saturation of the supercritical CO2 through a change in relative

volume or density. With these assumptions, the supercritical CO2 saturation following a change in pressure can be written as

$$S_{c,o} = S_c^- \frac{\rho_o}{\rho_f} \qquad \textbf{Eq. (50)}$$

where $S_{c,o}$ [-] is the saturation after the pressure change and ρ_f [kg·m^{-3}] is the density after the pressure change. The saturation after the pressure change will determine the fraction of supercritical CO2 that has become mobilized, X_{mobile} [-]. This relationship can be expressed as

$$X_{mobile} = \left(\frac{S_{c,o} - S_c^-}{S_{c,o}}\right) = \left(1 - \frac{\rho_f}{\rho_o}\right). \qquad \textbf{Eq. (51)}$$

This definition assumes that if the mobile supercritical CO2 were to migrate from its original region, it would leave behind residually trapped CO2 (at S_c^-) in the wake of its motion, as discussed in section 1. Using equation 50, the mobilized fraction of CO2 can also be expressed as a function of ρ_o and ρ_f, presented on the far right hand side of equation 51.

In general, the density of supercritical CO2 is a function of pressure and temperature, as shown in Figure 3 for $P_{crit} \leq P \leq 60\ MPa$ and three different temperatures, where P_{crit} is the critical pressure of CO2 ($\sim$7.3 MPa). We will assume that the temperature of the reservoir remains constant throughout all processes, given the large heat capacity of the rock bed and neglecting any heat flux across the boundary of the reservoir domain, such that density variations follow an isothermal path and are only a function of pressure.

Using equations 50 and 51 and the density data corresponding to a temperature

of 315 K shown in Figure 3, we plot $S_{c,o}/S_c^-$ and X_{mobile} as a function of pressure drop in Figure 11 (a) and (b), respectively, for three initial CO2 densities: $\rho_o = 600$ kg·m^{-3} (red), $\rho_o = 700$ kg·m^{-3} (blue), and $\rho_o = 800$ kg·m^{-3} (green). These three initial density conditions correspond to initial CO2 pressures (and approximate reservoir depths) of 10.2 MPa (1.04 km), 12.1 MPa (1.23 km), and 17.3 MPa (1.76 km), respectively, where the approximate reservoir depths are calculated assuming a hydrostatic pressure gradient from surface to reservoir.

The results in Figure 11 highlight two important aspects of remobilizing CO2. First, given the non-linearity of $\rho(P)$ in the supercritical regime, CO2 with a higher initial density (green) produces a lower mobile fraction of CO2 for a given pressure drop. Generally speaking, this would indicate that CO2 sequestered in deeper reservoirs is less susceptible to becoming remobilized after being trapped in residual pockets. Second, in all three of these scenarios, large pressure drops are required to remobilize even half of the CO2; this observation demonstrates the robustness of residual trapping mechanisms as a means of immobilizing CO2. Because sequestration sites are selected, in part, for their geologic stability, one would not expect large pressure fluctuations to occur from natural events, and it is therefore more likely that any pressure fluctuations would result in changing the CO2 saturation by only a small amount above S_c^-. As we will demonstrate, this last point becomes important when considering how best to model the relative permeability of CO2 and the consequences this choice has on the results of modeling the CO2 dynamics.

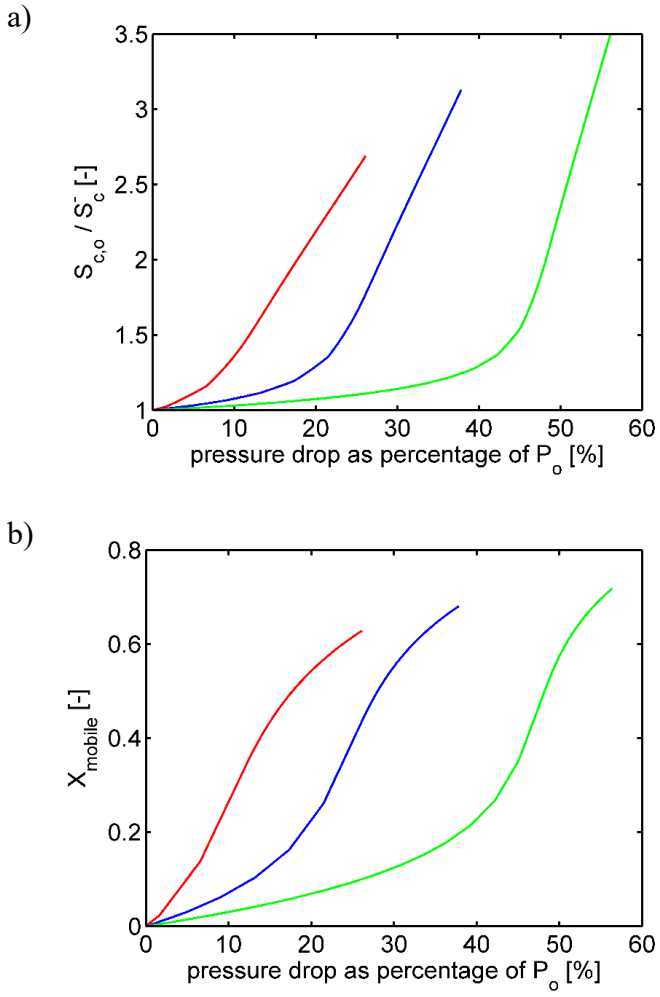


Figure 11. Remobilization of CO2 as a function of a decrease in pressure relative to a prescribed initial pressure. (a) Relative saturation, $S_{c,o}/S_c^-$, calculated using equation 50. (b) Mobile fraction, X_{mobile}, calculated using equation 51. In (a) and (b) the initial pressure, P_o, is 10.2 MPa (red), 12.1 MPa (blue), and 17.3 MPa (green). These values of initial pressure correspond to initial CO2 density values, ρ_o, of 600, 700, and 800 kg/m^3, respectively. The density of CO2 after depressurization, ρ_f, is determined from thermodynamic data (Linstrom and Mallard, 2011), given the final pressure. In all cases the temperature is assumed constant and equal to 315 K.

3.2.2 Scaling analysis of CO2 migration

Before proceeding to the analysis of remobilization, we present a brief scaling analysis related to the sequence of CO2 motion outlined in Figure 10b. Of particular interest is the time scale for CO2 motion in the vertical (z-)direction compared to the time scale for CO2 motion in the horizontal (r-)direction. In general, these time scales, τ, can be expressed as $\tau_i \sim L_{char,i}/u_i$ with $i = r, z$, where $L_{char,i}$ is the characteristic length in the i direction over which the CO2 travels and u_i is the characteristic fluid velocity in the i direction. In the z-direction, $L_{char} \sim h_o$, which is the thickness of the formation, and $u_z \sim (\kappa \Delta \rho g/\mu)$, given by the Darcy velocity where the pressure gradient driving the flow is a result of buoyancy acting on the density difference between CO2 and brine. The characteristic time is then given by $\tau_z \sim (h_o \mu)/(\kappa \Delta \rho g)$. In the horizontal direction, the choice of L_{char} is less obvious in the absence of a well-defined boundary. Here, we let L_{char} be the characteristic radius originally occupied by the CO2 plume, $L_{char} = R_o$. Using a lubrication approximation for the flow, the characteristic velocity in the r-direction is given by $u_r \sim (\kappa \Delta \rho g h_o)/2R_o$ (Hesse et al., 2008; Huber et al., 2016). Using these results, $\tau_r \sim (\mu R_o^2)/(\kappa \Delta \rho g h_o)$. The ratio of these two time scales is given by $\tau_r/\tau_z \sim R_o^2/h_1^2$. From geologic surveys, the thickness of many potential CO2 reservoirs, $h \sim 1 - 100$ meters (DOE, 2012; Dooley et al., 2005; Finley, 2005; Meyer, 2007; Steadman et al., 2006; Wickstrom et al., 2005). However, the volume of CO2 that may be injected at any given site is potentially large (thousands to millions of cubic meters), such that the extent of the plume, $R_o \sim 1$ km. Therefore, it is likely that in most sequestration reservoirs $\tau_r/\tau_z \gg 1$. Using the characteristic reservoir properties listed in Table 5, τ_z is approximately 2 years and τ_r is approximately 200,000 years (both of

which are much greater than the 79 day estimate for poroelastice diffusion, τ_p, discussed in section 2.1). This result is important because it suggests that the problem can be considered in two distinct steps: CO2 rising in the z-direction (Figure 10b ii), followed by CO2 spreading in the horizontal direction (Figure 10b iii). The first step can be analyzed as a one dimensional countercurrent flow, while the second step can be analyzed as a process where global transport of CO2 occurs in the horizontal but the distribution of CO2 saturation is in quasi-steady state in the z-direction.

3.2.3 Modeling the relative permeability of CO2 near the residual saturation limit

There are many functions that exist to model the relative permeability of individual fluids within a multiphase porous medium (eg. Corey, Brooks-Corey, Van Genuchten, Kozeny) (Alpak et al., 1999; Corey, 1954; Land, 1971; van Genuchten, 1980). These functions are similar in that they all contain one or more fitting parameters, whose values are determined empirically and are dependent on the characteristics of the porous medium as well as the fluids present. Here, we consider a natural porous medium (geologic reservoir) containing only two fluids: supercritical CO2 and brine. One commonly used model to describe the relative permeability of CO2 and brine in such a system is the Brooks-Corey (BC) model, which can be written as

$$k_c = k_c^+ \left(\frac{S_c - S_c^-}{S_c^+ - S_c^-} \right)^n \qquad \textbf{Eq. (52)}$$

and

$$k_b = k_b^+ \left(\frac{S_c^+ - S_c}{S_c^+ - S_c^-} \right)^m \qquad \textbf{Eq. (53)}$$

for the relative permeability of CO2, k_c [-], and brine, k_b [-], respectively. In both of these equations, S_c^-[-] is the minimum residual CO2 saturation, S_c^+ [-] is the maximum CO2 saturation, k_c^+ [-] is the maximum relative permeability of CO2, and k_b^+ [-] is the maximum relative permeability of brine. All of these variables, along with the "Corey exponents," n [-] and m [-], are fitting parameters, while the saturation of CO2, S_c, is the independent variable.

There are several advantages of the Brooks-Corey model. First, each relative permeability function is smooth and continuous on the open interval $S_c^- < S_c < S_c^+$. Second, these functions are easy to implement within a numerical solver provided the fitting parameters are known. Last and perhaps most importantly, these functions have been shown to fit experimental data very well for a broad range of geologic samples and CO2 saturation values.

Although the Brooks-Corey model is widely used, it may not be sufficiently accurate to capture the relative permeability of CO2 near the residual limit. Here, we propose an alternative model for the relative permeability of CO2, which we will call the Brooks-Corey-Hunt (BCH) model. The goal of this model is to match the BC model at moderate and high values of CO2 saturation while exhibiting permeability scaling laws near S_c^- that come from percolation theory; this modification will, except in rare circumstances, cause this model to deviate from the BC model at saturations close to S_c^-.

Percolation theory describes the connectivity and transport properties within a randomly connected network (Hunt, 2005; Larson et al., 1981; Stauffer, 1985). The theory considers a lattice consisting of empty sites that are randomly populated at some

probability, p. Any two neighboring sites that are populated are considered linked via the bond between them that acts as a pathway. At some sufficiently high value of p, a single pathway will exist that will span the entire lattice, at which point the lattice is said to be percolated. The value of p necessary to percolate the network is called the percolation threshold, p_c, and is dependent upon the geometry of the lattice. More importantly however, percolation theory provides a universal prediction for the transport properties and the correlation length of the connected pathways near the percolation threshold; for a given dimension of space and regardless of lattice structure, transport properties are found to scale as a power law of the quantity (p-p_c).

Percolation theory has been applied to models of two-phase flow in porous media by treating the porous medium as a network of pores and pore throats with randomly assigned radii (and thereby capillary pressures). When two immiscible fluids occupy pore space within the network, the capillary trapping of one phase in residual pockets throughout the domain is analogous to this phase losing a connected pathway throughout the lattice. This idea has been explored and discussed in great detail by several authors who compare many of the scaling laws defined by percolation theory to the results of numerical simulations of one fluid phase invading another within a pore network (Ferer et al., 2003; Hunt, 2005; Wilkinson and Willemsen, 1983). Percolation theory dictates that as one phase loses connectivity within a network, the relative permeability of this phase scales as $k \propto (S - S^{-})^{1.88}$, where S is the saturation of the phase losing connectivity and S^{-} is the residual saturation value of the phase within the network. There remains some uncertainty in the value of the exponent with a recent study finding its value to be approximately 2 (Hunt et al, 2014). Hunt (2005) has

incorporated the percolation scaling law into a model for the relative permeability of water in subsaturated soils and shown that the resulting predictions for the relative permeability near the residual water saturation are confirmed by some experimental soil data. Our hybridized model is based build on this work.

The proposed BCH model uses the following piece-wise function to describe the relative permeability of CO2:

$$k_{cp} = \begin{cases} a_p \left(\dfrac{S_c - S_c^-}{S_c^+ - S_c^-} \right)^{1.88} & S_c^- \le S_c \le S_{cx} \\ k_c^+ \left(\dfrac{S_c - S_{cp}^-}{S_c^+ - S_{cp}^-} \right)^{n_p} & S_{cx} \le S_c \le S_c^+ \end{cases} \qquad \textbf{Eq. (54)}$$

where, S_c remains the independent variable while S_c^-, S_c^+, and k_c^+ maintain their definitions as fitting parameters, consistent with their use in equations 52 and 53. Before defining the additional parameters in equation 54, we note three important properties of this BCH model. First, like the BC model, this function is smooth and continuous on the open interval $S_c^- < S_c < S_c^+$. Second, consistent with the purpose of this new model and with the scaling laws defined in percolation theory, the relative permeability scales as $k \propto (S - S^-)^{1.88}$ as the saturation of CO2 approaches S_c^-. Finally, as the CO2 saturation increases, the behavior of this function mimics that of the BC model. We call the CO2 saturation at which this function transitions from following the scaling laws of percolation theory to following the common BC model the cross over saturation, S_{cx} [-]:

$$S_{cx} = S_c^- + \frac{1}{\phi}\left[\frac{1.88(1-\phi)}{\dfrac{3}{3 - D_p} - 1.88} \right] \qquad \textbf{Eq. (55)}$$

where ϕ [-] is the porosity of the domain and D_p [-] is the dimensional connectivity of the pore network. Due to the complexity of natural pore networks like those found in rocks and soils, D_p cannot be derived and is treated as another fitting parameter in this model; although it is a physical requirement that its value be less than 3 and it has been shown to be near 2.96 for many soils (Hunt, 2005). Similarly, the exponent n_p [-] is also a fitting parameter and is used to capture the curvature of the relative permeability at large saturation values. Here, n_p has the same role as the Corey exponent n in equation 52. The other two parameters in equation 54, a_p and S_{cp}^-, are defined as

$$a_p = k_c^+ \left(\frac{S_c^+ - S_c^-}{S_{cx} - S_c^-}\right)^{1.88} \left(\frac{S_{cx} - S_{cp}^-}{S_c^+ - S_{cp}^-}\right)^{n_p} \quad \textbf{Eq. (56)}$$

and

$$S_{cp}^- = \frac{1}{1.88}\left[1.88 S_{cx} - n_p(S_{cx} - S_c^-)\right]. \quad \textbf{Eq. (57)}$$

These parameters are defined to ensure that the relative permeability function is smooth and continuous over the entire saturation range ($S_c^- < S_c < S_c^+$). We note that a_p and S_{cp}^- are not additional fitting parameters; they are defined by the other parameters of this model.

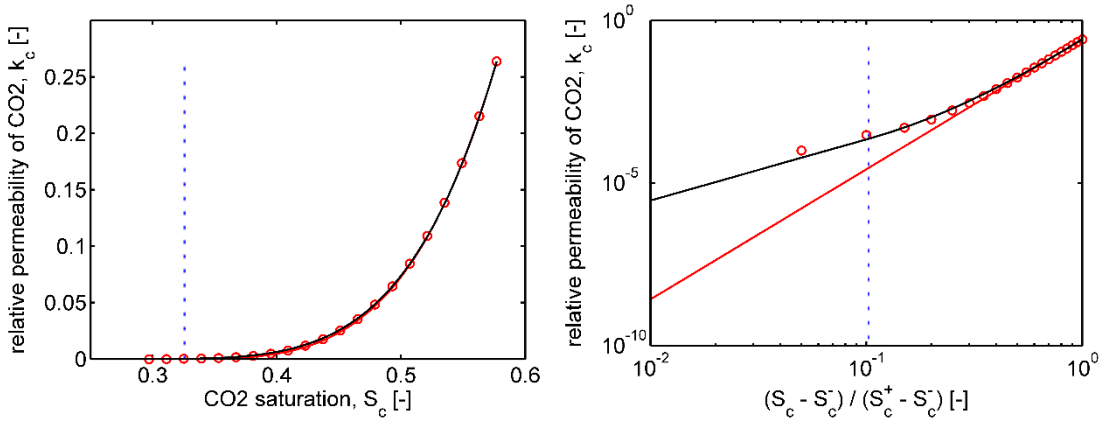
Comparing the BC and BCH models, the BC model contains four fitting parameters, (k_c^+, S_c^-, S_c^+, n), while our BCH model contains five, $\left(k_c^+, S_c^-, S_c^+, n_p, D_p\right)$. Generally speaking, one would expect an empirical model with more degrees of freedom to better fit experimental data. However, that expectation may not necessarily be met in this case because of the role that these parameters have within the model. In both models, k_c^+, S_c^-, and S_c^+ are used to define the bounds of the function such that k_c

varies from 0 to k_c^+ over the saturation range S_c^- to S_c^+. In the BC model, the Corey exponent n defines the curvature of the permeability function between these bounds, while in the BCH model, the exponent n_p defines the curvature of the permeability function only between the saturation values of S_{cx} and S_c^+. For saturation values below S_{cx}, the curvature of the BCH function is fixed by the exponent 1.88, set from percolation theory. The only role of the additional fitting parameter in the BCH model, D_p, is to define the value of S_{cx}. As a result, the BCH model may not fit experimental data better than the BC model if the data does not exhibit permeability which scales as $(S - S^-)^{1.88}$ near S_c^-. Theoretically, a sufficiently well connected porous medium should follow this scaling, but some natural porous media may not satisfy this constraint.

Demonstrating that CO2 relative permeability may scale as $(S - S^-)^{1.88}$ near S_c^- in real rock samples at temperatures and pressures similar to those found at reservoir depth is challenging because to do so requires several experimental data points near the saturation threshold below which no CO2 flows through the sample. Here, we present the raw experimental data from two different sandstones collected by Bachu, who has published several works summarizing the best-fit values of the fitting parameters used in the BC model for a large range of rock samples containing CO2 and brine (Bachu, 2013; Bennion et al., 2008; Bennion and Bachu, 2006). The experimental data, BC model, and BCH model for these two rock samples are presented in Figure 13, while the numerical values used for all fitting parameters in the BC and BCH models, along with the relevant rock properties reported by Bachu, are listed in Table 4. In both of these data sets, it is difficult to discern any differences between the BC and BCH models

at first glance as both fit their corresponding experimental data with an $R^2 > 0.999$. Furthermore, when plotted on linear-linear axes, the BC and BCH models appear to completely overlap one another (left plots in Figure 13(a) and (b)). However, when the same information is plotted on log-log axes with the x-axis rescaled as $(S_c - S_c^-)/(S_c^+ - S_c^-)$, it becomes apparent that the experimental data changes slope near S_c^- and that the BCH model agrees with the experimental data better than the BC model at these low saturation values (right plots in Figure 13(a) and (b)). For the Viking and Nisku formations shown here, we found the value of S_{cx} to be 0.326 and 0.262, respectively (dashed blue line in all plots), leaving only three data points below this saturation, one of which corresponds to $k_c(S_c = S_c^-) = 0$. We cannot claim that this experimental data confirms the permeability scaling, $k \propto (S - S^-)^{1.88}$, required by percolation theory, since too few data points lie within this data set at saturation values below the cross-over saturation threshold, S_{cx}. However, fitting this low saturation data with a generalized model $k \propto (S - S^-)^{\zeta}$, we find that values of ζ between 1.5 and 2 bound the data points below S_{cx} present in each data set and additional data points would be required to better constrain the best-fit model. These results motivate the use of our BCH model and in the remainder of this work we will compare this model to the BC model as it relates to answering the questions outlined in section 1 regarding the dynamics of remobilized CO2.

a)



b)

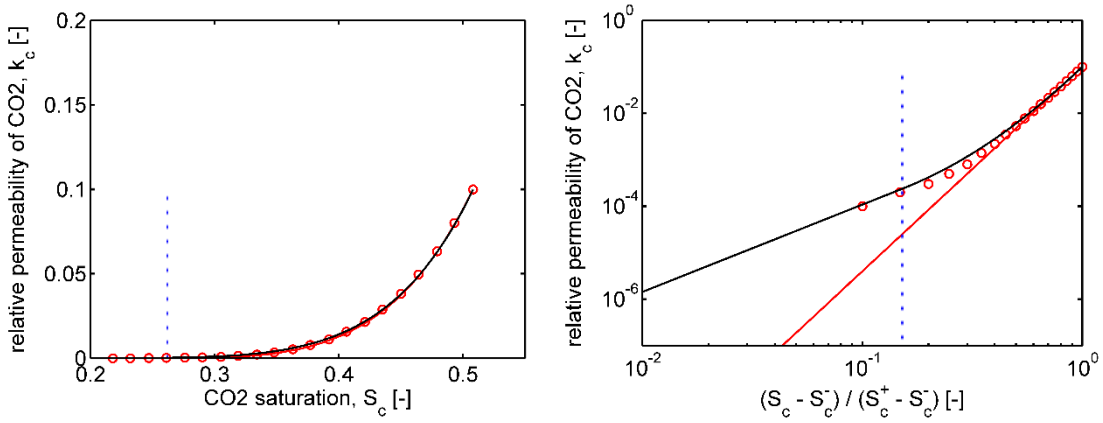


Figure 12. Comparison of Brooks-Corey (BC) and Brooks-Corey-Hunt (BCH) models with experimentally measured relative permeability of CO2 in rocks from two formations. (a-b) Experimental data (open circles) and best fits with the BC model (red curve, equation 52) and BCH model (black curve, equations 54 – 57), with marker to visually indicate the value of S_{cx} (dashed blue line), for rocks from the Viking formation (a) and Nisku formation (b) (Bachu, 2013; Bennion et al., 2008; Bennion and Bachu, 2006). In both (a) and (b), the best fit curves use the values listed in Table 4 and the results are plotted with linear axes (left) and on log-log axes (right).

			Viking Formation	Nisku Formation
best-fit values for BCH model		*porosity, ϕ [-]	0.195	0.114
	best-fit values for BC model	*exponent for relative permeability of CO2, n [-]	4.0	4.4
		*maximum relative permeability of CO2, k_c^+ [-]	0.2638	0.0999
		*residual saturation of CO2, S_c^- [-]	0.297	0.218
		*maximum saturation of CO2, S_c^+ [-]	0.577	0.508
		exponent for relative permeability of CO2, n_p [-]	4.66	5.93
		fractal dimensionality coefficient, D_p [-]	2.989	2.991

Table 4. Porosity data and best-fit values for the fitting parameters in equation 52 (BC model) and equations 54 – 57 (BCH model) based on relative permeability experiments involving the flow of CO2 and brine through core samples taken from the Viking and Nisku formations, from the work of Bachu. Items marked with (*) come directly from (Bachu, 2013).

3.3. Analysis of remobilized CO2 rising locally within the reservoir

3.3.1 Description

Based on the scaling analysis in section 2.3, we first analyze the local rise of remobilized CO2 within the reservoir independent of global CO2 migration in the horizontal direction. This motion can then be classified as two-phase, countercurrent flow in one dimension, where volume conservation requires that the upward motion of CO2 must occur simultaneously with brine flowing downward in the domain. We assume that a small decrease of pressure has occurred such that the local CO2 saturation is slightly above the minimum residual saturation and that this saturation is initially constant throughout the z-dimension, such that $S_c(t = 0, z) = S_{c,o}$. We seek to model the evolution of the CO2 saturation profile from this initial condition, the steady-state distribution of CO2 when motion ends, the time required for this process to occur, and the influence that the choice of relative permeability functions may have on the results.

3.3.2 Governing Equations

In a porous medium containing two or more fluid phases, the conservation of mass for a single phase, *i*, can be written as

$$\frac{\partial}{\partial t}[\phi \rho_i S_i] + \boldsymbol{\nabla} \cdot [\rho_i \boldsymbol{u}_{di}] = 0 \qquad \textbf{Eq. (58)}$$

where ϕ [-] is the porosity of the porous medium, ρ [kg·m^{-3}] is the density of the phase, S [-] is the phase saturation, and $\boldsymbol{u}_d$ [m·s^{-1}] is the Darcy velocity. In multiphase flow, the general form of the Darcy velocity for phase, *i*, is defined as

$$\boldsymbol{u}_{di} = \frac{\kappa k_i}{\mu_i}[(-\boldsymbol{\nabla} P_i) + \rho_i \boldsymbol{g}] \qquad \textbf{Eq. (59)}$$

where κ [m^2] is the intrinsic permeability of the porous medium, k [-] is the relative permeability of the phase, μ [Pa·s]is the viscosity of the fluid, P [Pa] is the fluid pressure, and $\boldsymbol{g}$ [$m \cdot s^{-2}$] is the acceleration due to gravity. As written, equation 58 ignores chemical reactions and the dissolution or evaporation of one phase into another. In this case, we assume that the brine within the reservoir is already saturated with CO2, hence no further dissolution will take place. In this section, we consider the one dimensional motion of CO2 and brine in the z-direction, parallel with the gravitational field, implying that the conservation of mass for CO2 can be written as

$$\frac{\partial}{\partial t}[\phi \rho_c S_c] + \frac{\partial}{\partial z}\left[\rho_c u_{dc,z}\right] = 0 \qquad \textbf{Eq. (60)}$$

and the mass conservation of brine can be written as

$$\frac{\partial}{\partial t}[\phi \rho_b S_b] + \frac{\partial}{\partial z}\left[\rho_b u_{db,z}\right] = 0\,, \qquad \textbf{Eq. (61)}$$

where the Darcy velocities are defined as

$$u_{dc,z} = \frac{\kappa k_c}{\mu_c}\left[\left(-\frac{\partial P_c}{\partial z}\right) - \rho_c g\right] \qquad \textbf{Eq. (62)}$$

and

$$u_{db,z} = \frac{\kappa k_b}{\mu_b}\left[\left(-\frac{\partial P_b}{\partial z}\right) - \rho_b g\right] \qquad \textbf{Eq. (63)}$$

for CO2 and brine, respectively.

In this analysis, we assume that the density of CO2 and brine remain constant, the reservoir porosity is homogeneous, and the intrinsic permeability is homogeneous and isotropic. Applying these assumptions and substituting in the Darcy velocities,

equations 60 and 61 can be written as

$$\phi \frac{\partial S_c}{\partial t} + \frac{\kappa}{\mu_c} \frac{\partial}{\partial z} \left[k_c \left[\left(-\frac{\partial P_c}{\partial z} \right) - \rho_c g \right] \right] = 0 \qquad \textbf{Eq. (64)}$$

and

$$\phi \frac{\partial S_b}{\partial t} + \frac{\kappa}{\mu_b} \frac{\partial}{\partial z} \left[k_b \left[\left(-\frac{\partial P_b}{\partial z} \right) - \rho_b g \right] \right] = 0 \,, \qquad \textbf{Eq. (65)}$$

respectively. These equations are directly coupled by recognizing that there are only two fluid phases in the medium and, therefore, by the definition of saturation, $S_c + S_b = 1$. However, these three equations are not solvable on their own because there are a total of four unknowns: S_c, S_b, P_c, and P_b. To close the equations, the pressure in the brine and CO2 can be related through the capillary pressure, P_{cap} [Pa], given by

$$P_{cap} = P_c - P_b \,. \qquad \textbf{Eq. (66)}$$

At the pore scale, the capillary pressure can be defined using the Young-Laplace equation,

$$P_{cap} = \sigma \left(\frac{1}{R_1} + \frac{1}{R_2} \right) \qquad \textbf{Eq. (67)}$$

where σ [N·m^{-1}] is the surface tension of the CO2-brine phase interface, and R_1, R_2 are the radii of curvature of the meniscus between the phases. However, coupling the CO2 and brine pressures at the pore scale in this manner becomes intractable, as it requires perfect knowledge of the pore geometry and the surface interactions of all fluids within the domain. Instead, it has been shown that the capillary pressure within a control volume containing a large number of pores can be modeled as a function of the wetting phase saturation through the expression

$$P_{cap}(S_b) = \left(\sigma\sqrt{\frac{\phi}{\kappa}}\right)\mathcal{J}(S_b) \qquad \textbf{Eq. (68)}$$

where $\mathcal{J}(S_b)$ is called the Leverett J-function; developed by Leverett in 1941 (Leverett, 1941). This function was developed after recognizing that capillary pressure curves in different samples of the same composition could be collapsed into a single function, $\mathcal{J}(S_b)$, by normalizing by the factor $\sigma\sqrt{\phi/\kappa}$. The form of $\mathcal{J}(S_b)$ can vary, but here, we will use a the following polynomial

$$\mathcal{J}(S_b) = a_{j1}(1 - S^*) + a_{j2}(1 - S^*)^2 + a_{j3}(1 - S^*)^3 \qquad \textbf{Eq. (69)}$$

where a_{j1}, a_{j2}, and a_{j3} are fitting parameters (1.417, -2.120, and 1.263, respectively) and S^* is the reduced brine saturation, given by

$$S^* = \frac{S_b - S_b^-}{1 - S_b^-} \qquad \textbf{Eq. (70)}$$

where S_b^- is the minimum residual saturation of brine. This function is the general equation used in PetraSIM to model sequestration reservoirs (PetraSIM, 2015). $\mathcal{J}(S_b)$ has been plotted for the domain $0.2 \leq S_b \leq 0.75$ in Figure 13; similar polynomial forms have been used to describe drainage of CO2 and brine in the Frio formation of sandstone (Silin et al., 2009). While the fitting parameter values may change or the functional form of the J-function may change depending on the type of rock being considered, two features of this function remain the same. First, $\mathcal{J}(S_b)$ has a minimum, non-zero value as S_b approaches S_b^+. Second, the value of $\mathcal{J}(S_b)$ tends to increase as S_b decreases. Both of these characteristics are related to the same concept: in the tertiary brine-CO2-rock system, brine is the wetting phase and therefore, if CO2 invades the porous medium, it will preferentially invade larger pores first (larger radii) because the capillary pressure

in these pores is lower, defined in equation 67. As CO2 continues to displace brine within a given region, it must displace brine from smaller pores (greater capillary pressure), which implies that $\mathcal{J}(S_b)$ increases as S_b decreases. The value of the Leverett J-function as S_b continues to decrease towards a minimum residual value, S_b^-, will vary depending the choice of $\mathcal{J}(S_b)$. While here we use a polynomial, often in literature, $\mathcal{J}(S_b)$ takes the form $(S_b - S_b^-)^{-\Lambda}$, where Λ is a fitting parameter (Golding et al., 2011; Larson et al., 1981). Using this latter type of function requires that $\mathcal{J}(S_b)$ approach infinity (infinite capillary pressure) as S_b approaches S_b^-. This functional form is motivated by observations that indicate brine becomes residually trapped during drainage and after this saturation is reached, increasing the CO2 pressure does not appreciably alter the saturation of brine. However, a pore network scale view of the process of residual trapping suggests that the brine continuous percolating network of brine filled pores will first lose connectivity by the invasion of CO2 into a pore of finite radius, so that the capillary pressure as S_b approaches S_b^- will be finite. Nevertheless, both forms of $\mathcal{J}(S_b)$ have the same qualitative characteristics over a broad range of brine saturations.

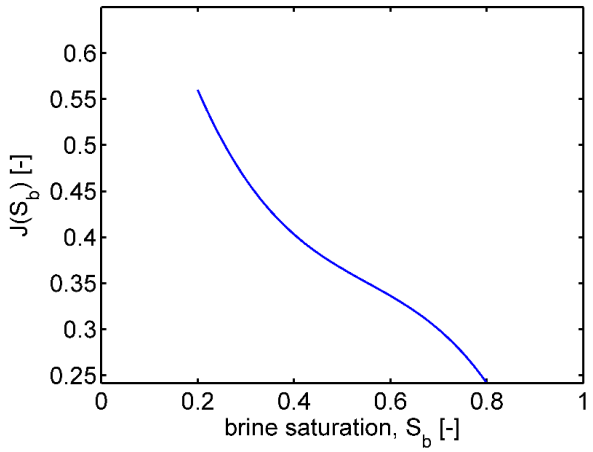


Figure 13. Leverett J-function, $\mathcal{J}(S_b)$, as a function of brine saturation, S_b, using equations 69 and 70.

Having introduced a Leverett J-function to model the capillary pressure as a function of brine saturation, we have now a closed set of equations to describe the dynamics of remobilized CO2. To solve, we first add equations 64 and 65, then recognize that $\partial(S_c + S_b)/\partial t = 0$. Combining this result with equation 66 and assuming no net flux of fluid across the domain boundaries, the pressure gradient in the CO2 can be expressed as

$$\left(-\frac{\partial P_c}{\partial z}\right) = \frac{1}{k_c + \eta k_b}\left(k_c \rho_c g + \eta k_b \rho_b g - \eta k_b \frac{\partial P_{cap}}{\partial z}\right) \qquad \textbf{Eq. (71)}$$

where η is the ratio of CO2 viscosity to brine viscosity, μ_c/μ_b. This pressure gradient can be substituted back into equation 64, combined with equation 68, and nondimensionalized to yield

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[k_c \frac{\eta f}{1 + \eta f}\left(1 - \frac{1}{Bo}\frac{d\mathcal{J}(S_c)}{dS_c}\frac{\partial S_c}{\partial \tilde{z}}\right)\right] = 0 \qquad \textbf{Eq. (72)}$$

which is the governing equation describing the evolution of remobilized CO2. In

equation 72, f is the ratio of brine relative permeability to CO2 relative permeability, k_b/k_c. The independent variables, t and z, have been nondimensionalized such that

$$\tilde{t} = t\frac{\kappa\Delta\rho g}{\phi\mu_c h_o} \qquad \textbf{Eq. (73)}$$

and

$$\tilde{z} = \frac{z}{h_o} \qquad \textbf{Eq. (74)}$$

respectively, where $\Delta\rho = (\rho_b - \rho_c)$ [kg·m^{-3}] and h_o [m] is the thickness of the sequestration reservoir. Equation 72 also introduces a Bond number, which is defined as

$$\mathrm{Bo} = \frac{g\Delta\rho_o h_o}{\sigma\sqrt{\phi/\kappa}} \qquad \textbf{Eq. (75)}$$

representing the ratio of the gravity force, causing the brine pressure to increase with increasing depth, and the capillary pressure, which limits the amount of brine pressure that is transmitted to the CO2 and limits the rise of the CO2.

At this juncture, we have derived a nondimensional equation (eq. 72) that governs the CO2 dynamics of interest, subject to appropriate boundary conditions. We note that equation 72 is mathematically equivalent to a nonlinear convection-diffusion equation with respect to CO2 saturation, where "convection" (first term in brackets) is driven by the density differences between the CO2 and brine and the "diffusion" (second term in brackets) of CO2 saturation results from capillarity because it is energetically favorable for brine to occupy the porous medium, hence there is the tendency of brine to displace the CO2 within the smallest pores of a given region. The importance of "diffusion" relative to "convection" is determined by the product of $\partial\mathcal{J}(S_c)/\partial S_c$ and

the Bond number, with larger Bond numbers resulting in less “diffusion” of saturation.

3.3.3 Description of simulations

We aim to understand how the initial saturation condition, $S_c(\tilde{t}=0,\tilde{z})$, Bond number, and choice of CO2 relative permeability function, k_c, effect the dynamics of remobilized CO2 described by equation 72. We solved this equation numerically using a forward-time flux difference technique for three different initial saturation conditions: $S_{c,o}/S_c^- = \{1.05, 1.10, 1.20\}$, where the initial CO2 saturation distribution was constant in the z-direction, $S_c(\tilde{t}=0,\tilde{z}) = S_{c,o}$. No flux boundary conditions were applied to the top and bottom of the reservoir to represent impermeable rock layers existing below and above the domain. For each value of $S_{c,o}/S_c^-$, equation 72 was solved using three different Bond numbers (0.05, 0.1, and 0.2). To provide consistent comparison, the parametric values listed in Table 5 were used in all simulations.

To explore the effects of k_c on the dynamics of the CO2 motion, at each value of $S_{c,o}/S_c^-$, several simulations were performed using the BC model, in which the CO2 Corey exponent, n, was varied over the range of 1.1 – 4. These simulations were then repeated using the BCH model, where the value of D_p was held constant and the value of n_p was determined to match the curvature of $k_c(S_c)$ from the BC model at the higher values of saturation.

The no flux boundary conditions require that CO2 mass, and volume, is conserved throughout the process. As discussed in section 1, the convection of mobile CO2 from a region leaves behind CO2 at a saturation equal to S_c^-. These two statements

allow us to write an equation to describe the conservation of CO2 saturation before and after motion occurs with the reservoir. This equation is given by

$$\int_0^1 S_c(\tilde{z}, \tilde{t} = 0)\, d\tilde{z} = \int_0^{\tilde{z}_b} S_c^-\, d\tilde{z} + \int_{\tilde{z}_b}^1 S_c(\tilde{z}, \tilde{t} = \infty)\, d\tilde{z} \qquad \textbf{Eq. (76)}$$

where $\tilde{z}_b$ [m] is the location in the reservoir below which the CO2 is completely immobile. Above $\tilde{z}_b$ the CO2 saturation must be greater than S_c^-. For a given initial condition, Bond number, and Leverett J-function, the steady state distribution of CO2 saturation within the entire domain can be found, a priori, by solving equation 76 for $\tilde{z}_b$ using the no-flux, steady state solution of equation 23, i.e.,

$$\frac{dS_c}{d\tilde{z}} = Bo\left(\frac{d\mathcal{J}(S_c)}{dS_c}\right)^{-1} \qquad \textbf{Eq. (77)}$$

for the saturation in $\tilde{z}_b \leq \tilde{z} \leq 1$.

Because the solution to equations 76 and 77 provide the steady state distribution of CO2 saturation, we can use this information to define a parameter, y, given by

$$y = \frac{\int_{\tilde{z}=0}^{\tilde{z}=1}\left(S_c(\tilde{z}, \tilde{t}) - S_c(\tilde{z}, \tilde{t} \to \infty)\right) d\tilde{z}}{\int_{\tilde{z}=0}^{\tilde{z}=1}\left(S_c(\tilde{z}, \tilde{t} = 0) - S_c(\tilde{z}, \tilde{t} \to \infty)\right) d\tilde{z}}. \qquad \textbf{Eq. (78)}$$

The value of y represents the unaccomplished decay of the dynamic solution towards its steady state result. Using this definition, each simulation was stopped when $y = 1/e^2$ (~0.135).

Parametric values used for the porous medium, CO2, and brine			
porosity, ϕ [-]	0.15	CO2 viscosity, μ_c [Pa·s]	1 x 10^{-4}
permeability, κ [m^2]	1 x 10^{-14}	CO2 viscosity:brine viscosity, η [-]	0.10
surface tension between CO2 and brine, σ [mN · m^{-1}]	20	characteristic density difference between brine and CO2, $\Delta\rho_o$ [kg·m^{-3}]	150
characteristic reservoir height, h_o [m]	1	Radius of the initial residually trapped plume, R_o [m]	1000
Parameters for Brooks-Corey and Brooks-Corey-Hunt relative permeability functions			
maximum relative permeability of CO2, k_c^+ [-]	0.5	maximum relative permeability of brine, k_b^+ [-]	1.0
maximum saturation of CO2, S_c^+ [-]	0.80	residual saturation of brine, S_b^- [-]	0.20
residual saturation of CO2, S_c^- [-]	0.25	brine Corey exponent, m [-]	2.0
fractal dimensionality coefficient in Brooks-Corey-Hunt model, D_p [-]	2.99		
Parameters for Leverett $\mathcal{J}$-function			
$\mathcal{J}$-function coefficient, a_{j1}, [-]	1.417	$\mathcal{J}$-function coefficient, a_{j2}, [-]	-2.120
$\mathcal{J}$-function coefficient, a_{j3}, [-]	1.263		

Table 5. Parametric values used in all analysis sections of this chapter, unless otherwise noted.

3.3.4 Discussion and results

Before solving the dynamics of the remobilized CO2, we first consider the effects that initial saturation condition, $S_{c,o}$, and choice of Bond number, Bo, have on the steady state distribution of CO2 saturation in the reservoir in a state of mechanical equilibrium where buoyancy and capillary pressure gradient are in balance; this steady state involves neither local flow nor dissipation. Figure 14 depicts three different initial saturation scenarios, $\mathrm{S_{c,o}}/\mathrm{S_c^-}$ = 1.05, 1.10, and 1.20, with steady state solutions corresponding to three different Bond numbers (0.05, 0.10, and 0.20) shown for each initial saturation condition. These CO2 saturation profiles were found by solving equations 76 and 77. In all solution profiles, there exists a region at the bottom of the reservoir ($z/h_o = 0$) where the CO2 saturation has returned to the minimum residual value, S_c^-. The solution exhibits a discontinuity of slope at the z-location, $\tilde{z}_b$, where this minimum saturation region ends and the mobile CO2 region begins. Holding the Bond number constant, $\tilde{z}_b$ decreases as the initial mobile fraction of CO2, $\mathrm{S_{c,o}}/\mathrm{S_c^-}$, is increased; this dependence is a simple consequence of the conservation of CO2 volume within the domain. More notably, for a fixed value of $S_{c,o}$, $\tilde{z}_b$ increases as the Bond number increases. This result is expected because larger Bond numbers indicate that the buoyancy forces driving the CO2 upward are larger relative to the capillary forces that act to redistribute brine throughout the domain. In the limit of $Bo \to \infty$, the steady state saturation distribution will become a step function consisting of a region of S_c^+ at the top of the domain and a region of S_c^- below (black lines in Figure 14). In such a scenario, capillary forces would be negligible, and the equation governing the dynamics of the S_c field (equation 72) would become a wave equation, solvable using the method of

characteristics in a manner similar to that we have previously described (Huber et al., 2016).

Having found the steady state CO2 saturation profiles for the $S_{c,o}$ and Bo numbers of interest, we now turn to the dynamic evolution of the saturation field governed by equation 72. Although the steady state solution is independent of relative permeability, as neither k_c or k_b appear in equations 76 or 77, the relative permeability is expected to affect the dynamics, and it is here that we investigate the differences between the BC and BCH relative permeability models.

First, we consider the initial condition, $S_{c,o}/S_c^- = 1.10$ and $\mathrm{Bo} = 0.1$. Using the parameters listed in Table 5 and a Brooks-Corey exponent $n = 3$, we determined the evolution of the CO2 saturation by solving equation 72 using both the BC and BCH models of relative permeability and allowing the CO2 distribution to approach steady state ($y = 1/e^2$, defined by equation 78). The CO2 saturation profiles at several values of y are presented in Figure 15. In each plot, the red line corresponds to the steady state solution, while the evolving CO2 saturation profile is represented by the blue and green lines, corresponding to the use of the BC and BCH relative permeability models, respectively. These results indicate that the choice of relative permeability model does little to affect the shape of the saturation profile at a given value of y. However, the time required to reach a particular saturation distribution varies greatly between the two relative permeability models. The value of $\tilde{t}$ for each profile is noted in each plot insert in Figure 15 (written as nt). For this case, the BC model predicts the dynamics of CO2 motion to occur in a time span that is an order of magnitude slower than the dynamics predicted using the BCH model.

While Figure 15 presents the CO2 profiles at four instances of time and unaccomplished decay (y), Figure 16 presents the complete evolution of unaccomplished decay of each CO2 profile. From this data, we see that the differences in the dynamics predicted using the BC and BCH models grow as the saturation profile reaches steady state. At early times, the BC model predicts dynamics that are 2 – 3 times slower than the BCH model, while, when $y < 0.5$, the dynamics differs even more dramatically. Under these conditions, the majority of CO2 in the lower region of the reservoir has migrated to the upper, mobile region. The remaining mobile CO2 in the lower region of the domain is at a saturation near S_c^-; this saturation regime corresponds to the greatest difference in relative permeability between the BC and BCH models, as discussed in section 2.2. It is therefore unsurprising that, as the CO2 saturation approaches steady state in this case, the time required for CO2 motion to occur differs greatly when using the BC and BCH relative permeability models.

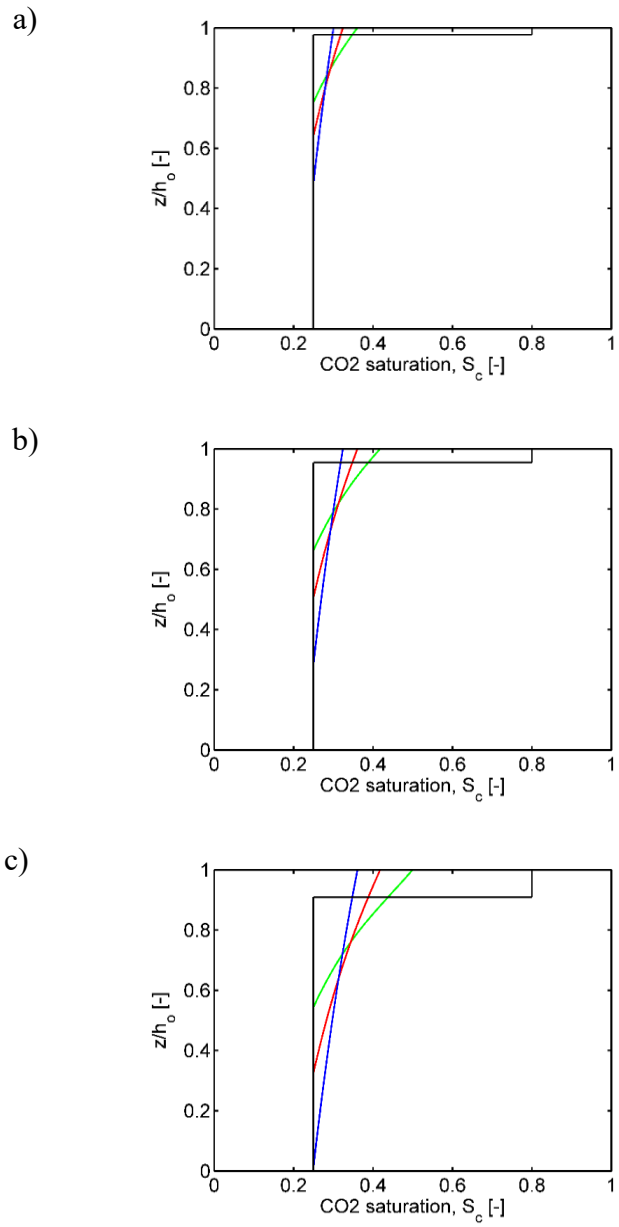


Figure 14. Steady-state, vertical distribution of CO2 saturation determined by solving equations 76 and 77 using the Leverett $\mathcal{J}$-function described by equations 69 and 70, assuming an initial remobilized fraction: (a) $S_{c,o}/S_c^- = 1.05$, (b) $S_{c,o}/S_c^- = 1.10$, and (c) $S_{c,o}/S_c^- = 1.20$. In all plots, Bo $= 0.05$ (blue), Bo $= 0.10$ (red), Bo $= 0.20$ (green), and Bo $= \infty$ (black).

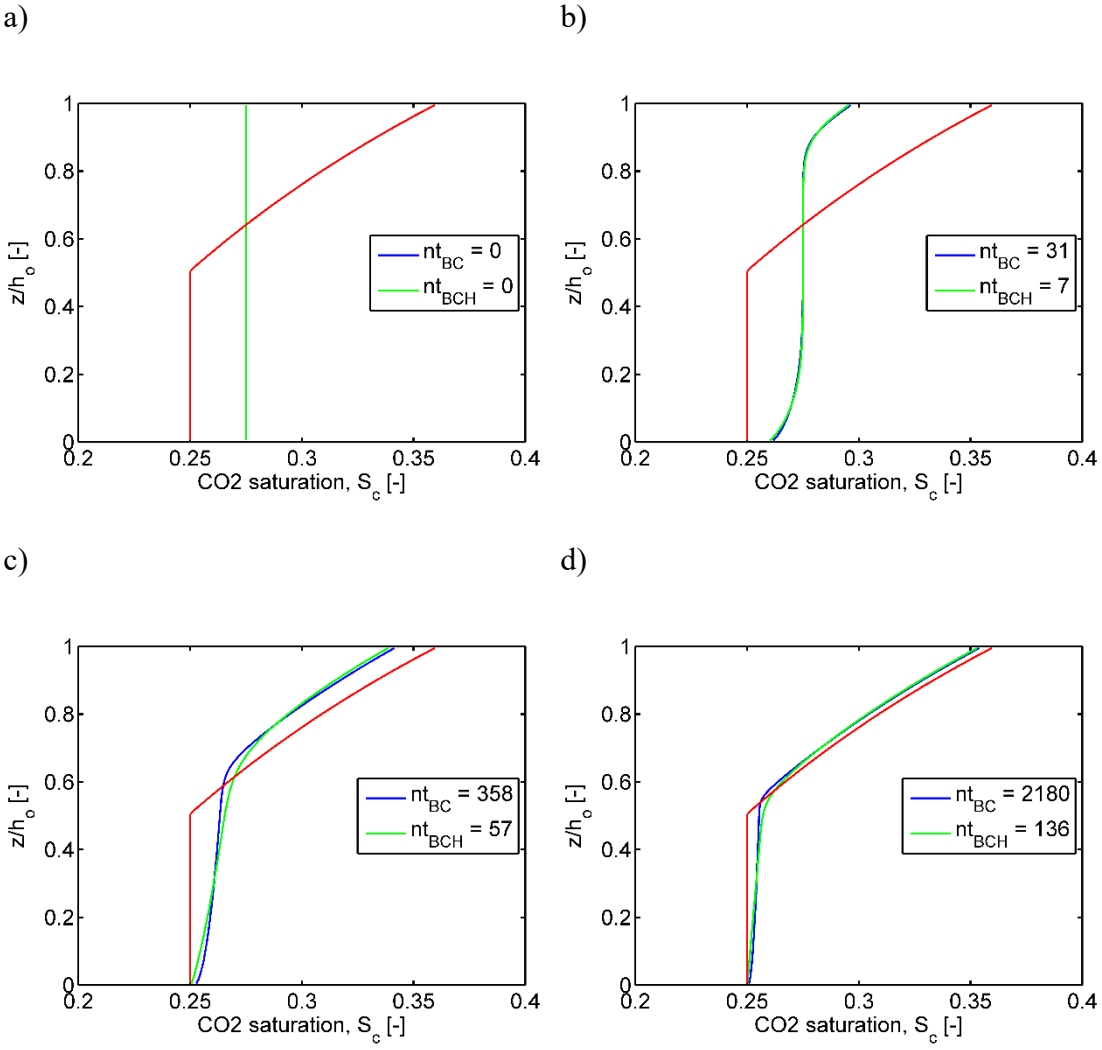


Figure 15. Evolution of the CO2 saturation using the Brooks-Corey model and Brooks-Corey-Hunt model for CO2 relative permeability at the values of unaccomplished decay: (a) $y = 1$, which corresponds to the initial condition, (b) $y = 0.90$, (c) $y = 0.37$, and (d) $y = 0.13$. This evolution assumed an initial saturation condition, $S_{c,o}/S_c^-$, of 1.10. In all plots, the Bond number is 0.10, the Brooks-Corey CO2 exponent (n) is 3.0, the Brooks-Corey-Hunt CO2 exponent (n_p) is 3.17, and the steady-state CO2 saturation profile (determined by solving equations 69 and 70) is the red curve.

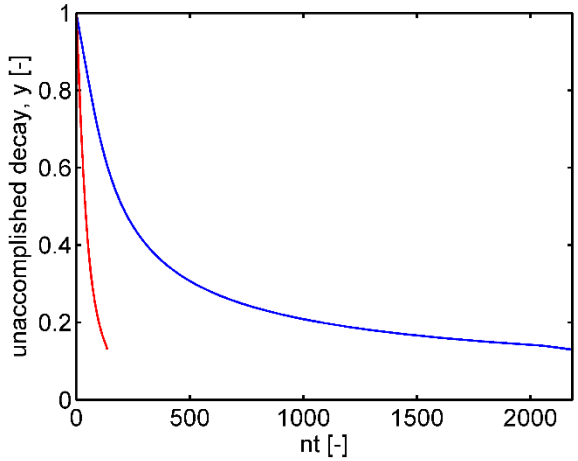


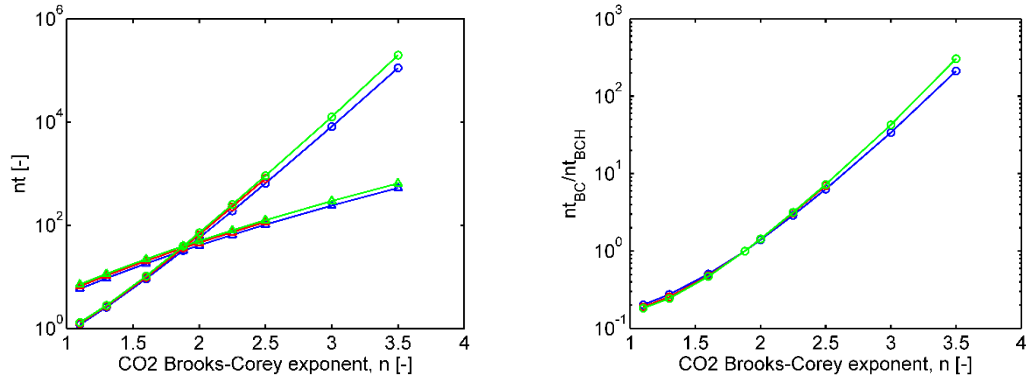
Figure 16. The unaccomplished decay of the CO2 saturation profile (y), as defined in equation 78, as a function of nondimensional time, as defined in equation 73, for the initial condition $S_{c,o}/S_c^- = 1.10$ and $\mathrm{Bo} = 0.10$ using the Brooks-Corey relative permeability model (blue) compared to the Brooks-Corey-Hunt model (red). The Brooks-Corey CO2 exponent (n) is 3.0 and the Brooks-Corey-Hunt CO2 exponent (n_p) is 3.17.

The results above all correspond to the use of the Corey exponent $n = 3$ in defining the BC relative permeability. This value represents an average Corey exponent found using experimental data of CO2 and brine flowing through sandstones (Bachu, 2013). However, relative permeability and best fit Corey exponent varies by rock sample and here we repeated the above analysis for Corey exponents ranging from 1.05 to 4. For each value of n, we determined the nondimensional time required for the CO2 saturation profile to reach an unaccomplished decay of $y = 1/e^2$ for three different initial saturation conditions and three different Bond numbers; Figure 17 presents these results, which demonstrate that the predicted dynamics are a strong function of Corey exponent and a weak function of initial condition and Bond number. As expected, the BC and BCH models predict identical results when $n = 1.88$, which is the permeability

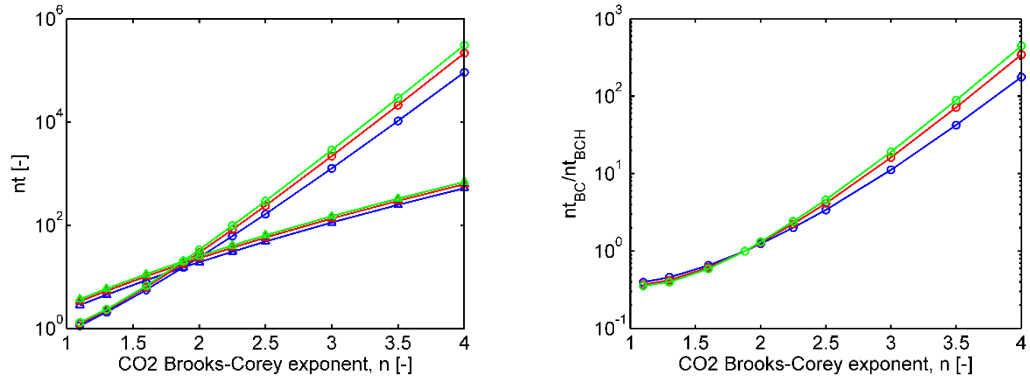
exponent imposed by the BCH model at saturation values near S_c^-. When $n > 1.88$, the BCH model predicts CO2 motion to occur more rapidly than the BC model, consistent with the $n = 3$ results presented above. These differences in the predicted time to reach steady state grow as the value of n increases, as shown in the ratio ($\tilde{t}_{\mathrm{BC}}/\tilde{t}_{\mathrm{BCH}}$) in the right hand column of Figure 17. For example, for $n = 4$ the time required to reach steady state is 2 – 3 orders of magnitude slower when using the BC model compared to 1 – 2 orders of magnitude slower for $n = 3$. However, for values of $n < 1.88$, the opposite trend occurs; the BCH model predicts slower dynamics than the BC model. Collectively, these results indicate that CO2 motion is predicted to occur faster for smaller values of n and is a function of the curvature of the relative permeability function near S_c^-.

With respect to the effects of Bond number and initial condition, the results in Figure 17 indicate that the initial volume of remobilized CO2 has a greater impact on the non-dimensional rise time. However, it is important to note that mapping these results back to dimensional values, using equation 73, involves the Bond number; namely, this transformation can be written as $t = \tilde{t}\phi\mu_c h_o^2/\left(\kappa\sigma \mathrm{Bo}\sqrt{\phi/\kappa}\right)$. As a result, the Bond number influences the dynamics in all of these cases, despite having little effect on the dimensionless rise time in the smallest initial volume condition (Figure 17a). While relative permeability and Corey exponent clearly exhibit the greatest influence on the rise time in these cases, the effects of Bond number and initial condition are also important. From the results shown here, differences in Bond number or initial condition may change the predicted dimensional rise time by nearly a factor of 10, for constant Corey exponent and choice of relative permeability model.

a)



b)



c)

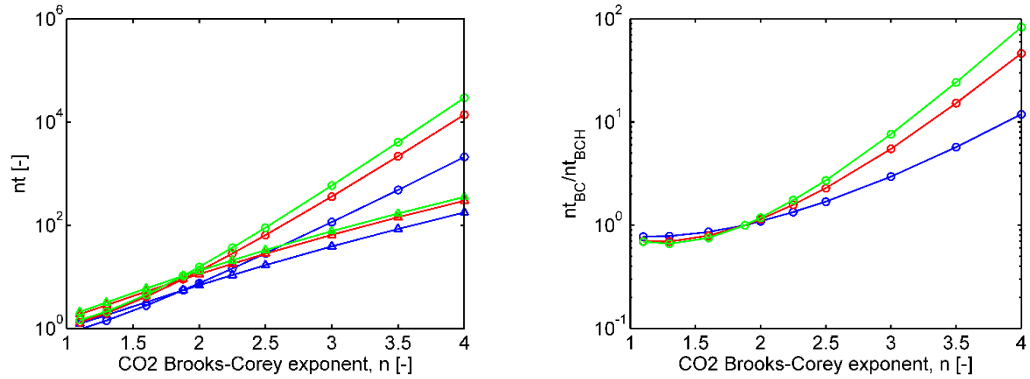


Figure 17. The nondimensional time required for the CO2 saturation profile, $S_c(\tilde{z}, \tilde{t})$, to reach the near-steady-state condition defined in equation 78 ($y = 1/e^2$) starting from an initial CO2 saturation condition given by: (a) $S_{c,o}/S_c^- = 1.05$, (b) $S_{c,o}/S_c^- = 1.10$,

and (c) $S_{c,o}/S_c^- = 1.20$. In (a-c) the nominal nondimensional time (left) is shown using the Brooks-Corey model for CO2 relative permeability (circles) and the Brooks-Corey-Hunt model for CO2 relative permeability (triangles) as a function of the Brooks-Corey exponent for CO2. The ratio of the nondimensional time, defined as $\tilde{t}_{BC}/\tilde{t}_{BCH}$, of these results are shown on the right. In all plots, $\mathrm{Bo} = 0.05$ (blue), $\mathrm{Bo} = 0.10$ (red), and $\mathrm{Bo} = 0.20$ (green).

3.4. Analysis of remobilized CO2 spreading radially within the reservoir

After mobile CO2 rises and fills the reservoir beneath the caprock it will spread horizontally in the domain due to buoyancy forces between the CO2 and brine. This global migration of CO2 can be problematic because it increases the risk of the plume encountering a fault in the caprock and leaking towards the surface. Therefore, it is important to consider the dynamics of this process to predict the extent of CO2 migration and also the time scales associated with this type of motion.

From the scaling analysis presented in section 2.3, the characteristic time for CO2 to rise vertically within the domain is estimated to be much smaller than the characteristic time for CO2 to migrate horizontally ($\tau_z \ll \tau_x$). Again, this conclusion is reached by assuming (1) the primary force driving the dynamics is the buoyancy between the CO2 and brine phases (which acts in the vertical direction) and (2) the geometry of the reservoir has a large aspect ratio with the vertical dimension being much smaller than the horizontal dimensions (which is generally true in layered strata). Given this difference in time scales, we will analyze the radial spreading of CO2 under the assumption that the CO2 is in quasi-steady state in the vertical direction throughout the dynamics of the horizontal motion.

3.4.1 Governing Equations

We would like to derive a simplified model of the horizontal spreading of the CO2 plume which captures the important dynamics of the migration process. Starting again from equation 58 and assuming constant porosity and fluid density, the

conservation of mass for the CO2 phase can be written as

$$\phi \frac{\partial S_c}{\partial t} + \frac{1}{r}\frac{\partial}{\partial r}\left[r u_{dc,r}\right] + \frac{\partial}{\partial z}\left[u_{dc,z}\right] = 0\,. \qquad \textbf{Eq. (79)}$$

First, we average equation 79 in the z-direction and nondimensionalize, recognizing that $\frac{1}{h_o}\int_0^{h_o}\frac{\partial}{\partial z}\left[u_{dc,z}\right]\,dz = 0$ due to the no flux condition at the boundaries between the reservoir and the surrounding impermeable layers. The average conservation equation is

$$\frac{\partial \bar{S}_c}{\partial \tilde{t}_r} + \frac{1}{\tilde{r}}\frac{\partial}{\partial \tilde{r}}\left[\overline{\tilde{r}\mathcal{D}\left(S_c^* + \bar{S}_c\frac{\partial S_c^*}{\partial \bar{S}_c}\right)}\frac{\partial \bar{S}_c}{\partial \tilde{r}}\right] = 0\,. \qquad \textbf{Eq. (80)}$$

In this equation, the nondimensional radial position is given by

$$\tilde{r} = \frac{r}{R_o}\,. \qquad \textbf{Eq. (81)}$$

The nondimensional time is now given by

$$\tilde{t}_r = \frac{\kappa \Delta\rho g h_o}{\phi \mu_c R_o^2}\,t, \qquad \textbf{Eq. (82)}$$

which is related to $\tilde{t}$ by $\tilde{t}_r = \tilde{t}R_o^2/h_o^2$. The overbar indicates an average over vertical positions defined for any property A by:

$$\bar{A}(\tilde{r},\tilde{t}) = \int_0^1 A(\tilde{r},\tilde{z},\tilde{t})\,d\tilde{z}\,. \qquad \textbf{Eq. (83)}$$

The variable effective diffusivity, $\mathcal{D}$ [-], defined as

$$\mathcal{D} = k_c\left(\frac{-f\eta}{1+f\eta}\right)\frac{1}{\mathrm{Bo}}\frac{d\mathcal{J}}{dS_c}, \qquad \textbf{Eq. (84)}$$

and S_c^* [-], which is the quasi-steady state saturation scaled by the average saturation whose relationship to S_c and $\bar{S}_c$ is defined by

$$S_c(\tilde{r},\tilde{z},\tilde{t}_r) = S_c^*(\tilde{z},\bar{S_c}) * \bar{S_c}(\tilde{r},\tilde{t}_r)\,. \qquad \textbf{Eq. (85)}$$

Performing these steps, maps the saturation field, $S_c(\tilde{r},\tilde{z},\tilde{t}_r)$, onto an averaged field, $\bar{S_c}(\tilde{r},\tilde{t}_r)$. Equation 80 defines the conservation of this average CO2 saturation within the domain. We note that unlike the equation governing the CO2 saturation dynamics in the z-direction (Eq. 72), dynamics in the radial direction are driven purely by capillary forces without a convective term present, since the fluid convection is due to gravity (buoyancy) and does not act in the radial direction. While the first term in equation 80 is relatively simple to understand (the time rate of change of the vertically-averaged CO2 saturation at some location, $\tilde{r}$), the second term in this equation is more complicated because the manner in which $\bar{S_c}$ diffuses in the domain strongly depends upon the details of the vertical distribution of CO2 within each region. In section 3, we analyzed the vertical migration of CO2 in a region that we assumed contained immobile CO2 everywhere while the remobilized fraction of CO2 convected to the top of the reservoir. This assumption, that the region of interest contained CO2 at S_c^- everywhere, implies that after remobilization, the vertically-averaged CO2 saturation of that region would correspond to a unique steady state CO2 saturation profile, and therefore, this known profile could be used to evaluate the diffusive term in equation 80. However, in general, $\bar{S_c}$ cannot be mapped to a unique vertical distribution of CO2, and as a result, solving equation 80 for this averaged field still requires knowledge of the vertical distribution of CO2 for all $\tilde{r}$ at the current $\tilde{t}_r$.

To simplify this model, we must make an important assumption about the location of CO2 invading into any region of the reservoir. Namely, we will assume that migrating CO2 enters the top of any region first and will invade downward to some

location $\tilde{z}_b$. By making this assumption, we restrict the location of mobile CO2 to only the top of any given region and thus the vertical distribution of this mobile CO2 will be described by the steady state distribution profile, solved for in section 3. This assumption also implies that for a given average value of mobile CO2, there exists a unique (and known) corresponding vertical distribution of CO2, where mobile CO2 is defined as

$$S_{cm} = \begin{cases} S_c(r,z,t) - S_c^- & S_c > S_c^- \\ 0 & S_c \leq S_c^- \end{cases} . \qquad \textbf{Eq. (86)}$$

Substituting this definition of mobile CO2 into equation 80 yields

$$\frac{\partial \bar{S}_{cm}}{\partial \tilde{t}_r} + \frac{1}{\tilde{r}} \frac{\partial}{\partial \tilde{r}} \left[\tilde{r} \overline{\mathcal{D} \left(S^*_{cm} + \bar{S}_{cm} \frac{\partial S^*_{cm}}{\partial \bar{S}_{cm}} \right)} \frac{\partial \bar{S}_{cm}}{\partial \tilde{r}} \right] = -S_c^- H \left(\frac{d}{d\tilde{t}_r} (\tilde{z}_b - \tilde{z}_{\max}) \right), \qquad \textbf{Eq. (87)}$$

where H is the Heaviside step function, $\tilde{z}_b$ is the current depth to which mobile CO2 invades the domain at that $\tilde{r}$ location, $\tilde{z}_{\max}$ is the maximum depth that CO2 has occupied at that $\tilde{r}$ position over all previous times, and

$$S_{cm}(\tilde{r}, \tilde{z}, \tilde{t}_r) = S^*_{cm}(\tilde{z}, \bar{S_c}) * \bar{S}_{cm}(\tilde{r}, \tilde{t}_r) . \qquad \textbf{Eq. (88)}$$

Comparing equation 87 to equation 80, the terms on the left hand side now track the time dependent change and local diffusion of the mobile portion of CO2 while the new sink term on the right hand side describes the loss of mobile CO2 due to the growth of the immobile, residually trapped CO2 within any given region. The Heaviside step function limits the applicability of the sink term only to situations where the vertical extent of the plume exceeds the maximum depth ever to have been occupied by the CO2 in the past. By making this assumption about how mobile CO2 invades a given region,

we are able to separate the problem into two primary variables, $\bar{S}_{cm}$, which is the average saturation of mobile CO2 within a given region and is independent of hysteretic information, and $\tilde{z}_{\max}$, which contains all of the hysteretic information regarding the residually trapped saturation of CO2. From these assumptions, the current mobile plume depth, $\tilde{z}_b$, is only a function of $\bar{S}_{cm}$, since each value of $\bar{S}_{cm}$ corresponds to a unique vertical steady state profile, $S_c(\tilde{z}, \tilde{t} \to \infty)$, determined from the solution to equation 72.

3.4.2 Description of Simulations

Since we assume that residually trapped CO2 has been remobilized and has risen locally within the reservoir until the saturation profile has reached near steady state in the z-direction, our analysis here focuses on the time required for this remobilized CO2 to spread horizontally within the reservoir. Following the analysis in section 3.3, given a particular initial value of $S_{c,o}/S_c^-$ and Bo number, we consider a reservoir containing a region of CO2 initially within a 1 km radius of the original injection well. We assume that the CO2 saturation profile in the z-direction is the steady state solution to equation 72 and that this saturation profile exists everywhere within the domain containing CO2. The vertically averaged mobile saturation distribution becomes the initial condition for the analysis of CO2 spreading horizontally. From this initial condition, equation 87 is solved subject to no-flux boundary conditions at $\tilde{r} = 0$ and $\tilde{r} \to \infty$, meaning that the CO2 and brine are in countercurrent flow, since each fluid is assumed incompressible.

We solve equation 87 numerically using a finite-volume scheme by first discretizing the domain in the radial direction. Each node in the $\tilde{r}$ domain stores a value

of $\bar{S}_{cm}$ and $\tilde{z}_{\max}$. The initial condition for $\tilde{z}_{\max}$ is applied to the domain by setting $\tilde{z}_{\max}(\tilde{r} \leq 1) = 1$ and $\tilde{z}_{\max}(\tilde{r} > 1) = 0$, since we have assumed that, within the original CO2 plume, trapped CO2 at S_c^- exists everywhere, and, outside the original CO2 plume, only brine occupies the pore space. The initial average mobile CO2 condition is applied to the domain by setting $\bar{S}_{cm}(\tilde{r} \leq 1) = \bar{S}_{cm,o}$ and $\bar{S}_{cm}(\tilde{r} > 1) = 0$, where $\bar{S}_{cm,o}$ is determined from the vertical, steady state solution to equation 72 using the specified values of $S_{c,o}/S_c^-$ and Bo number for that particular case. Like the vertical rise analysis in section 3, in this study of radial spreading we vary the initial CO2 saturation condition, $S_{c,o}/S_c^- = \{1.05, 1.10, 1.20\}$, Bond number, $\mathrm{Bo} = \{0.05, 0.10, 0.20\}$, and Brooks-Corey CO2 exponent, $n = 1.1 - 4$. These simulations were then repeated using the BCH model, where the value of D_p was held constant and the value of n_p was determined to match the curvature of $k_c(S_c)$ from the BC model at the higher values of saturation. To provide consistent comparison, the parametric values listed in Table 5 were used in all simulations.

Once the initial condition in the domain is set, the profile of $\bar{S}_{cm}$ is evolved in discrete time steps. At each time step, the change in $\bar{S}_{cm}$within each finite volume element is determined by the net inward flux of mobile CO2 at the boundaries of the volume element (diffusive term in equation 87), less any CO2 becoming trapped within that volume element due to an increase in $\tilde{z}_{\max}$. In this manner, the value of $\tilde{z}_{\max}$ is also updated at each node.

To demonstrate the application of this model, we present in Figure 18 the evolution of the CO2 saturation profile within two neighboring finite volume elements in the reservoir: one inside the original plume (left box plot in Figure 18a-d) and one

outside the original plume (right box plot in Figure 18a-d). The initial state of the reservoir is depicted in Figure 18a, with remobilized CO2 concentrated near the caprock of the domain, left, and only brine in the adjacent region, right. As the system is allowed to flow, the CO2 will invade the brine region, and as per our assumptions, this invasion will be confined to the top of the domain, such that invading CO2 will exist within a depth no greater than $\tilde{z}_b$ (marked with the dashed blue line in Figure 18a-d) of the neighboring volume element. As this invasion progresses, $\tilde{z}_{\max}$ (marked with the dashed black line in Figure 18a-d) within the brine region is updated, shown in Figure 18b. At later time, as the mobile plume continues to spread globally, the local depth of the mobile CO2 layer, $\tilde{z}_b$, decreases while the value of $\tilde{z}_{\max}$ remains unchanged, shown in Figure 18c. In the limit of infinite time, all mobile CO2 will be converted back to residually trapped CO2 and the value of $\tilde{z}_{\max}$ will identify the spatial arrangement of the trapped CO2 within the domain, shown in Figure 18d.

Similar to our definition of unaccomplished decay, y, in the analysis of vertical migration presented in section 3, here we define a stopping criterion based upon the local reduction of $\bar{S}_{cm}$ at the center of the plume, y_r, given by

$$y_r = \frac{\bar{S}_{cm}(\tilde{r} = 0, \tilde{t}_r)}{\bar{S}_{cm}(\tilde{r} = 0, \tilde{t}_r = 0)} \ . \qquad \textbf{Eq. (89)}$$

In this analysis of radial spreading, all numerical solutions were stopped when $y_r = 0.05$.

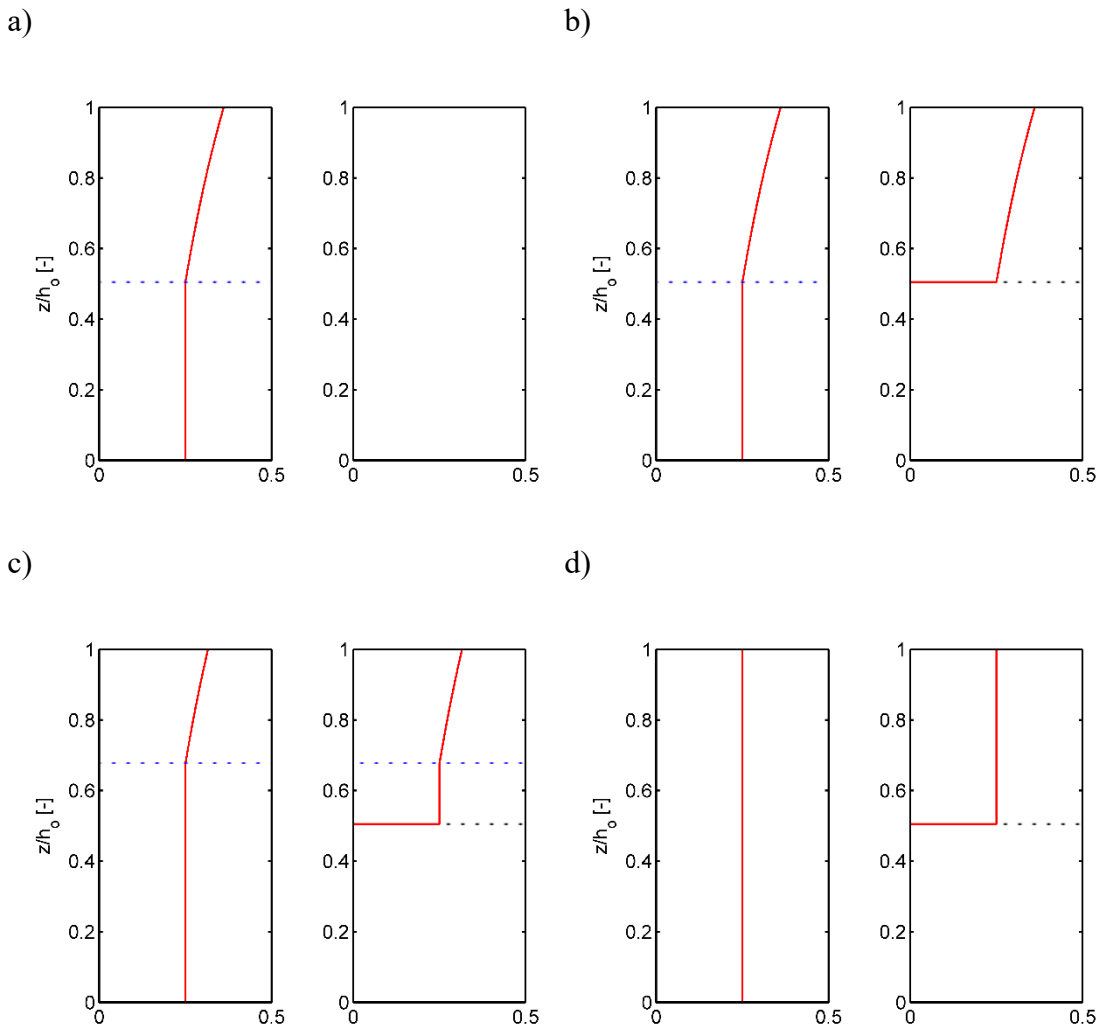


Figure 18. The evolution of the vertical steady state CO2 saturation profiles (red curves) near the initial boundary between the mobile plume, located at $1 - \Delta\tilde{r}/2$ (a-d, left), and the portion of the domain originally filled only with brine, located at $1 + \Delta\tilde{r}/2$ (a-d, right), as radial spreading occurs. (a) The initial state, $\tilde{t}_r = 0$. (b) At early times, mobile CO2 is invading the once brine-filled portion of the reservoir and in this region, $d/d\tilde{t}_r[\tilde{z}_b - \tilde{z}_{\max}] > 0$. (c) At later times, the mobile portion of the plume has thinned in this region, and the location of $\tilde{z}_b$ is receding, while $\tilde{z}_{\max}$ remains at its greatest positive value. (d) As $\tilde{t}_r \to \infty$, all mobile CO2 has been converted back to residually trapped CO2, such that $\tilde{z}_b = 0$ throughout the domain. In all plots the x-axis is the value of CO2 saturation, S_c, the location of $\tilde{z}_b$ is denoted with the dashed blue line, and the location of $\tilde{z}_{\max}$ is denoted with the dashed black line.

3.4.3 Discussion and results

Following the results in section 3.4, we first present the dynamics of remobilzed CO2 for a few individual cases of system conditions. Figure 19, 12, and 13 depict the evolution of $\bar{S}_{cm}$ as well as the corresponding locations of $\tilde{z}_b$ and $\tilde{z}_{\max}$ for Bond numbers of 0.05, 0.10, and 0.20, respectively. In all three figures, the initial volume of remobilization is given by $S_{c,o}/S_c^- = 1.10$ and the Brooks-Corey exponent used here was $n = 3$. As discussed in section 3, the Bond number plays a critical role in determining the extent to which mobile CO2 exists throughout the vertical direction, where smaller Bond number indicates that mobile CO2 is spread out vertically within the domain and larger Bond number indicates the mobile CO2 becomes concentrated beneath the caprock with a larger saturation. As a result, the maximum radial position of the plume is greater with increasing Bond number. We expect this trend because more extensive radial plume migration is necessary to access enough volume to residually trap the mobile CO2 at higher Bond numbers when it is concentrated in a thin layer at the top of the reservoir. We also note that in comparing these three cases, the nondimensional time required to reach the saturation profile depicted in plot (d) of each figure increases with increasing Bond number. Based on the idea that higher Bond number corresponds to higher and therefore more mobile CO2 concentrations near the top of the reservoir, this trend is counterintuitive. However, given the definitions of Bond number and $\tilde{t}$, the dimensional time can be expressed as $t = (\tilde{t}/\mathrm{Bo}) * \phi\mu_c R^2/\left(\kappa\sigma\sqrt{\phi/\kappa}\right)$, and so, for these cases of $\mathrm{Bo} = 0.05, 0.10, \text{and } 0.20$, the dimensional time required for plume motion, using the characteristic reservoir property values in Table 5, is summarized in Table 6. These dimensional results reveal that,

indeed, higher Bond number does correlate with faster plume migration, at least when using the BC relative permeability model. In using the BCH model however, the dimensional time required for migration is nearly independent of Bond number. This difference can be explained by the fact that during the plume spreading, regardless of Bond number, the CO2 saturation is essentially always near the residual saturation limit in these cases and as a result, the total CO2 plume mobility predicted by the BCH model is a weaker function of CO2 saturation compared to the BC model in this case of Brooks-Corey CO2 exponent $n = 3$.

	Time [years]	
	BC model	BCH model
$\mathrm{Bo} = 0.05$	2.85×10^7	5.42×10^6
$\mathrm{Bo} = 0.10$	1.97×10^7	5.53×10^6
$\mathrm{Bo} = 0.20$	1.37×10^7	5.53×10^6

Table 6. The dimensional time required to achieve the $\overline{S_{c_{mob}}}$ distribution depicted in Figure 19d, Figure 20d, and Figure 21d. The dimensional times listed were calculated using the nondimensional time found in simulation and the dimensional characteristic parameters listed in Table 5.

The dynamics in these cases also share several similarities to the dynamics of the vertical CO2 migration, shown in Figure 15. First, in both the vertical migration and in the radial spreading, the CO2 saturation profiles corresponding to the Brooks-Corey (BC) and Brooks-Corey-Hunt (BCH) relative permeability functions are very similar to one another for the same point of unaccomplished decay. Recall that these two models

of CO2 relative permeability only differ slightly when the local value of CO2 saturation is near the residual limit. Despite the qualitative similarities in the evolution of $\bar{S}_{cm}$ in these cases, the time required to achieve each particular saturation distribution is vastly different depending on the relative permeability model. Near the end of plume motion (plot (d) in Figures 11, 12, and 13) the model using BC relative permeability requires an amount of time that is approximately one order of magnitude greater than the model using BCH relative permeability. To better quantify how this time difference between the two relative permeability models occurs, the time required for each saturation profile to spread is shown in Figure 22 for the case of $\mathrm{Bo} = 0.10$ (corresponding to the saturation evolution in Figure 20). From this data, we find that at early times ($y_r > 0.6$), the two models predict similar dynamics of the plume spreading. However, at later stages of the plume spreading ($y_r < 0.6$), when the saturation of CO2 approaches the residual value throughout the domain, the time differences between these models grow dramatically, which is a consequence of how the BC and BCH model differ near S_c^-.

Another important, but subtle, difference between the saturation profiles predicted using the BC and BCH models is the behavior of the leading edge of mobile CO2 invading into the far-field of the reservoir. To examine this effect, Figure 23 presents the maximum radial position of mobile CO2 as a function of time, again for the case of $\mathrm{Bo} = 0.10$ (corresponding to the saturation evolution in Figure 20). The BCH model predicts the leading edge of CO2 to invade faster than the BC model. This result is not very surprising, since both models yield similar CO2 saturation profiles over the whole domain and the BCH model yields a much faster reduction in mobile CO2 at the center of the plume. However, the faster invasion rate predicted by the BCH

model also corresponds to a region of entrained trapped CO2 which is smaller than the residually trapped CO2 predicted by the BC model. This in turn corresponds to CO2 invading further into the reservoir in the BCH model than in the BC model. This result could be of great importance for reservoir management, since greater plume migration increases the risk of CO2 leaking from the domain.

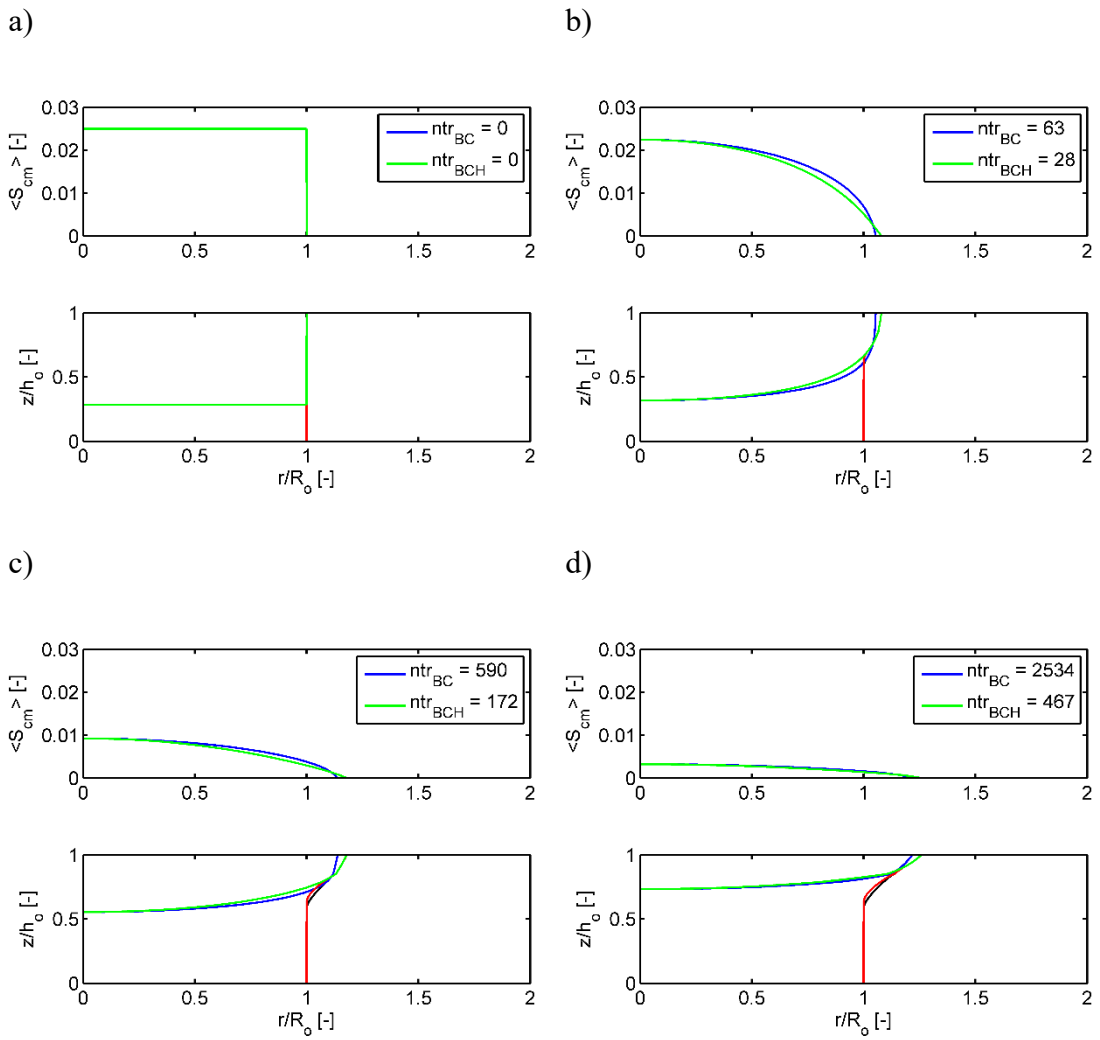


Figure 19. Comparison of the radial spreading of mobile CO2 using the Brooks-Corey (BC) model and Brooks-Corey-Hunt model (BCH) for CO2 relative permeability for the initial condition $S_{c,o}/S_c^- = 1.10$, Bo $= 0.05$, BC CO2 exponent (n) is 3.0, and BCH CO2 exponent (n_p) is 3.17. (a-d) The top plot is the current profile of $\bar{S}_{cm}$ using the BC model (blue curve) and the BCH model (green curve). The bottom plot depicts the location of the current boundary between mobile and trapped CO2, $\tilde{z}_b$ (blue curve-BC model, green curve-BCH model), and the maximum extent of the CO2 plume over all previous time, $\tilde{z}_{\max}$ (black curve-BC model, red curve-BCH model). (a) Initial condition, $y_r = 1$, (b) $y_r = 0.90$, (c) $y_r = 0.37$, (d) $y_r = 0.13$.

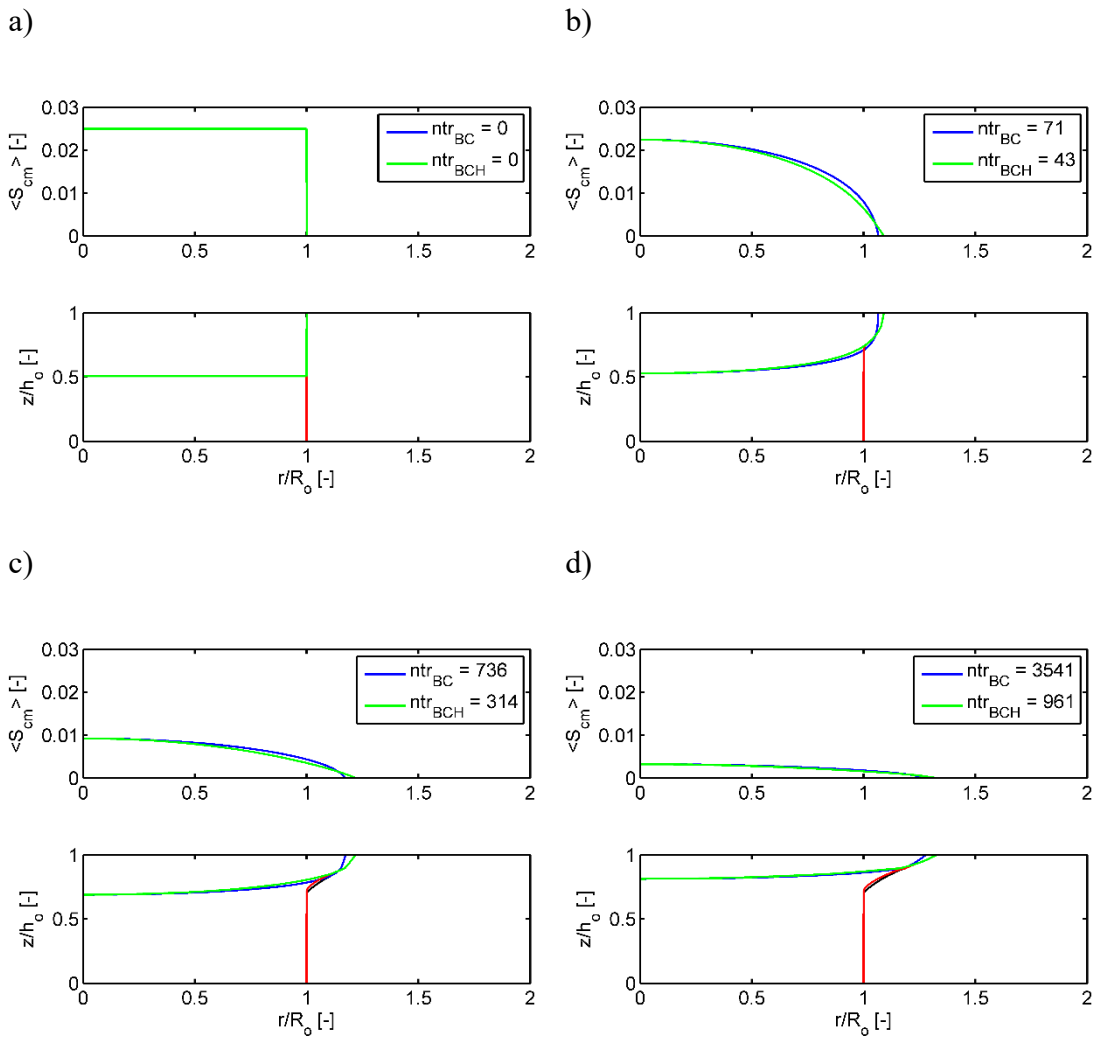


Figure 20. Comparison of the radial spreading of mobile CO2 using the Brooks-Corey (BC) model and Brooks-Corey-Hunt model (BCH) for CO2 relative permeability for the initial condition $S_{c,o}/S_c^- = 1.10$, Bo $= 0.10$, BC CO2 exponent (n) is 3.0, and BCH CO2 exponent (n_p) is 3.17. (a-d) The top plot is the current profile of $\bar{S}_{cm}$ using the BC model (blue curve) and the BCH model (green curve). The bottom plot depicts the location of the current boundary between mobile and trapped CO2, $\tilde{z}_b$ (blue curve-BC model, green curve-BCH model), and the maximum extent of the CO2 plume over all previous time, $\tilde{z}_{\max}$ (black curve-BC model, red curve-BCH model). (a) Initial condition, $y_r = 1$, (b) $y_r = 0.90$, (c) $y_r = 0.37$, (d) $y_r = 0.13$.

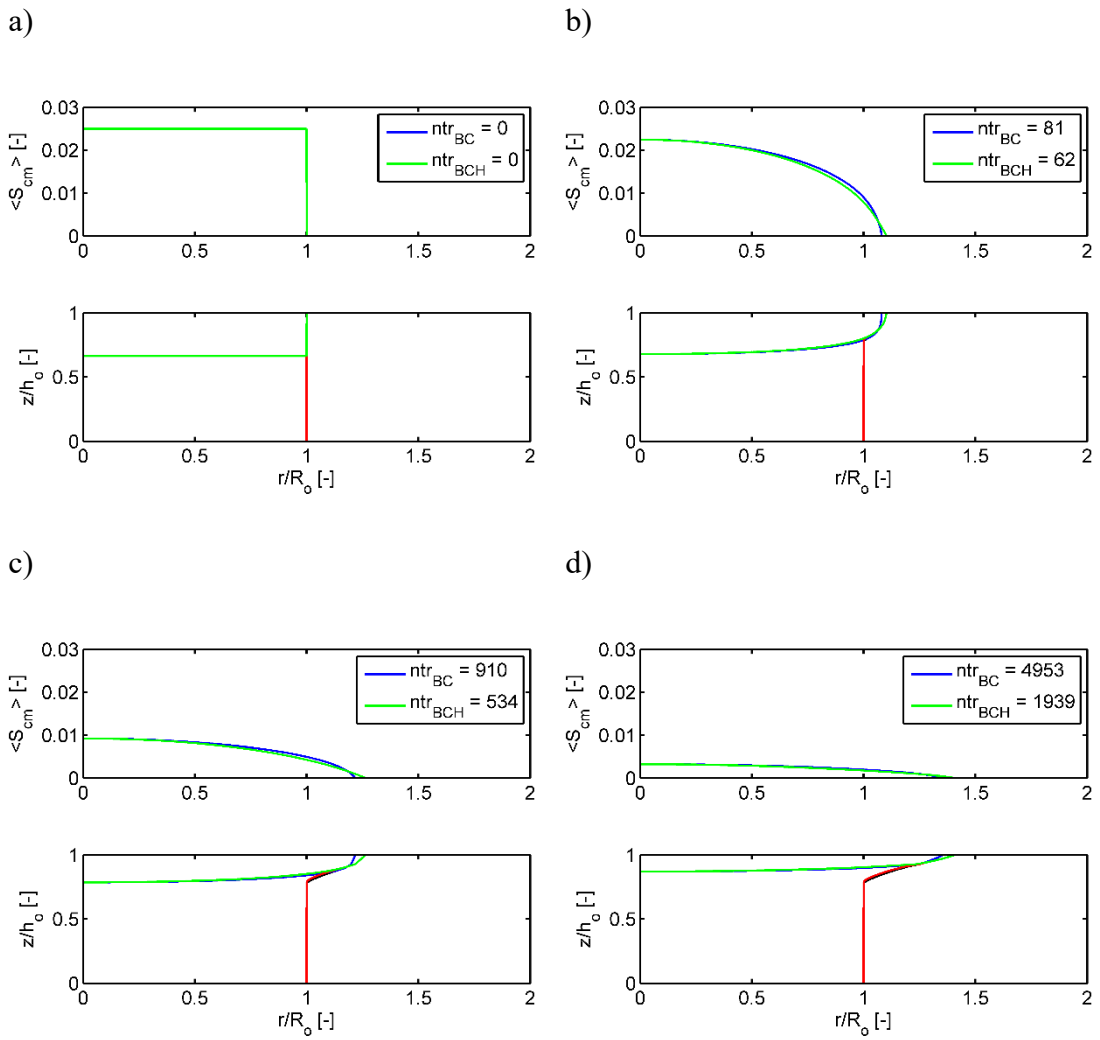


Figure 21. Comparison of the radial spreading of mobile CO2 using the Brooks-Corey (BC) model and Brooks-Corey-Hunt model (BCH) for CO2 relative permeability for the initial condition $S_{c,o}/S_c^- = 1.10$, Bo $= 0.20$, BC CO2 exponent (n) is 3.0, and BCH CO2 exponent (n_p) is 3.17. (a-d) The top plot is the current profile of $\bar{S}_{cm}$ using the BC model (blue curve) and the BCH model (green curve). The bottom plot depicts the location of the current boundary between mobile and trapped CO2, $\tilde{z}_b$ (blue curve-BC model, green curve-BCH model), and the maximum extent of the CO2 plume over all previous time, $\tilde{z}_{\max}$ (black curve-BC model, red curve-BCH model). (a) Initial condition, $y_r = 1$, (b) $y_r = 0.90$, (c) $y_r = 0.37$, (d) $y_r = 0.13$.

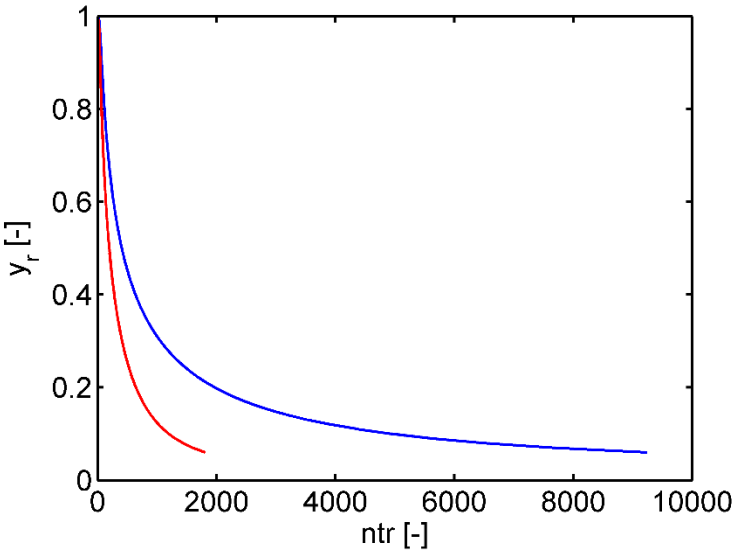


Figure 22. The average mobile CO2 remaining at the center of the plume ($\tilde{r} = 0$) normalized by the initial average mobile CO2 saturation as a function of nondimensional time, $\tilde{t}_r$, for the initial condition $S_{c,o}/S_c^- = 1.10$ and $\mathrm{Bo} = 0.10$ using the Brooks-Corey relative permeability model (blue) compared to the Brooks-Corey-Hunt model (red). The Brooks-Corey CO2 exponent (n) is 3.0 and the Brooks-Corey-Hunt CO2 exponent (n_p) is 3.17.

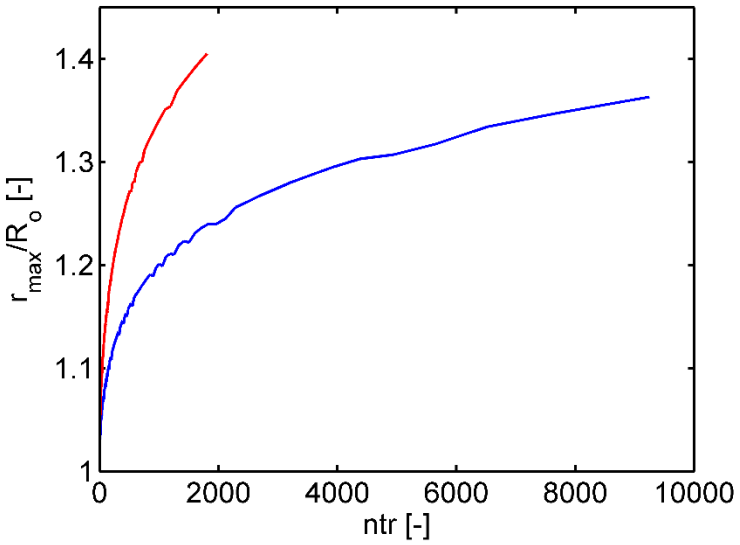


Figure 23. The maximum radial distance of mobile CO2, $\tilde{r}_{\max}$, within the reservoir as a function of the nondimensional time, $\tilde{t}_r$, required to achieve this migration, for the initial condition $S_{c,o}/S_c^- = 1.10$ and $\mathrm{Bo} = 0.10$ using the Brooks-Corey relative permeability model (blue) compared to the Brooks-Corey-Hunt model (red). The Brooks-Corey CO2 exponent (n) is 3.0 and the Brooks-Corey-Hunt CO2 exponent (n_p) is 3.17.

While the results in Figures 10 – 14 correspond to a single case of BC exponent and initial condition, the plots in Figure 24 and Figure 25 summarize the results of all cases studied (BC exponent $n = \{1.5, 1.88, 2, 2.5, 3, 4\}$, initial condition $S_{c,o}/S_c^- = \{1.05, 1.10, 1.20\}$, and Bond number $\mathrm{Bo} = \{0.05, 0.10, 0.20\}$). In Figure 24, we plot the nondimensional time required for the mobile CO2 saturation at the center of the plume to decrease to a value given by $y_r = 0.05$ (defined in equation 89). These results are similar in nature to the dynamics of vertical rise shown in Figure 17. In both migration directions, the time required for migration to occur is shorter using the BC relative permeability model when the BC exponent, n, is less than 1.88. When the BC exponent is equal to 1.88, then the BC and BCH relative permeability models predict identical migration times. For all BC CO2 exponents greater than 1.88, the BC model predicts

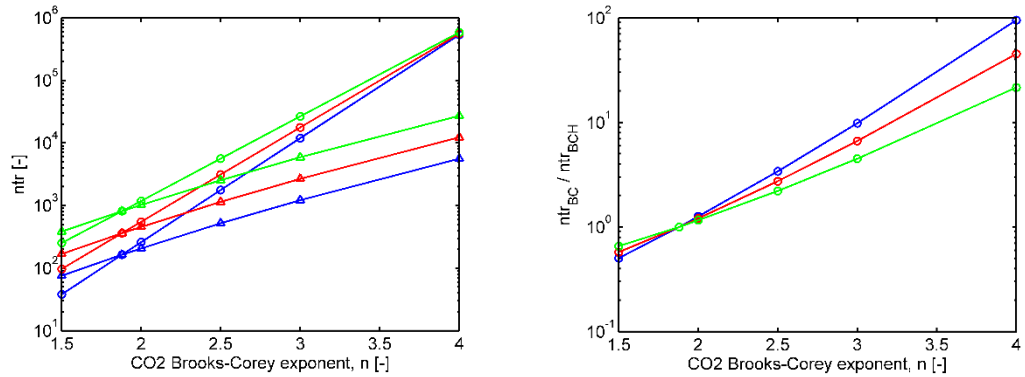
migration times that are much longer than the migration times predicted using the BCH model. In the case of the lowest initial fraction of remobilized CO2, $S_{c,o}/S_c^- = 1.05$, and highest BC CO2 exponent, the difference in migration time predicted by the BC and BCH models is two orders of magnitude.

Here, the importance of Bond number can vary depending on the relative permeability model used and the CO2 Corey exponent. Using the BCH model for relative permeability, larger Bond number results in more time required for plume motion, independent of the CO2 Corey exponent. However, using the BC model, the influence of Bond number diminishes at larger CO2 Corey exponent. While the Bond number is an indicator of how large the maximum CO2 saturation will become when the saturation distribution is in steady state, ultimately, as the plume radially spreads, the overall thickness of the mobile region decreases and the maximum CO2 saturation at any given location approaches S_c^-. As this mobile CO2 region diminishes, the time scale for radial migration becomes dominated by the relative permeability near this S_c^- limit, and as a result, the effect of Bond number becomes insignificant relative to the diminishing CO2 mobility predicted by the BC model at these low saturations. However, since the BCH model predicts greater CO2 mobility near the S_c^- limit and fixes the power law exponent for the dependence of relative permeability on saturation near this limit to be equal to 1.88 regardless of the Corey exponent for greater saturations of CO2, the effect of Bond number does not diminish at larger Corey exponent.

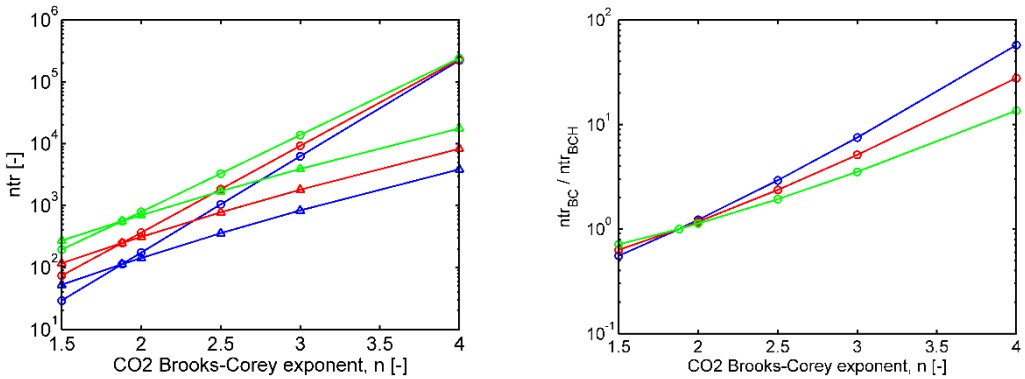
Finally, in Figure 25, we present the maximum radial distance of the mobile CO2 within the domain at the nondimensional time when $y_r = 0.05$. As one might expect, when the degree of remobilization is higher (more mobile CO2 is initially

present), the CO2 plume radially spreads further into the reservoir (Figure 25c vs. Figure 25a). In addition, higher Bond number (greater fraction of mobile CO2 accumulates near the upper boundary of the reservoir) also results in the remobilized CO2 migrating further into the domain. The nonobvious trend is that the BCH model predicts the CO2 plume will invade further into the far-field than the BC model for all Corey exponents greater than 1.88 (Figure 25). As discussed above, the greater CO2 mobility of the BCH model yields a slightly smaller volume of trapped CO2 in the wake of plume motion and, as a result, a slightly larger volume fraction of CO2 remains mobile and continues to invade further into the domain. Together, these results suggest that if the relative permeability of CO2 follows the BCH model more closely than the BC model, then remobilized CO2 will migrate faster and further than predicted by the standard Brooks-Corey model.

a)



b)



c)

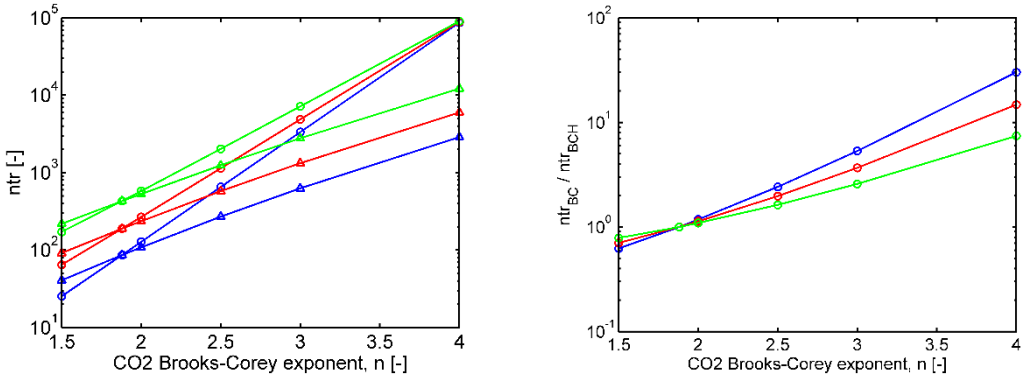


Figure 24. The nondimensional time, $\tilde{t}_r$, required for the average mobile CO2 saturation profile, $\bar{S}_{cm}(\tilde{r}=0,\tilde{t})$, to reach the near-steady-state condition defined in equation 89

($y_r = 0.05$) starting from an initial CO2 saturation condition given by: (a) $S_{c,o}/S_c^- = 1.05$, (b) $S_{c,o}/S_c^- = 1.10$, and (c) $S_{c,o}/S_c^- = 1.20$. In (a-c) the nominal nondimensional time (left) is shown using the Brooks-Corey model for CO2 relative permeability (circles) and the Brooks-Corey-Hunt model for CO2 relative permeability (triangles) as a function of the Brooks-Corey exponent for CO2. The ratio of the nondimensional time, defined as $\tilde{t}_{rBC}/\tilde{t}_{rBCH}$, of these results are shown on the right. In all plots, $\mathrm{Bo} = 0.05$ (blue), $\mathrm{Bo} = 0.10$ (red), and $\mathrm{Bo} = 0.20$ (green).

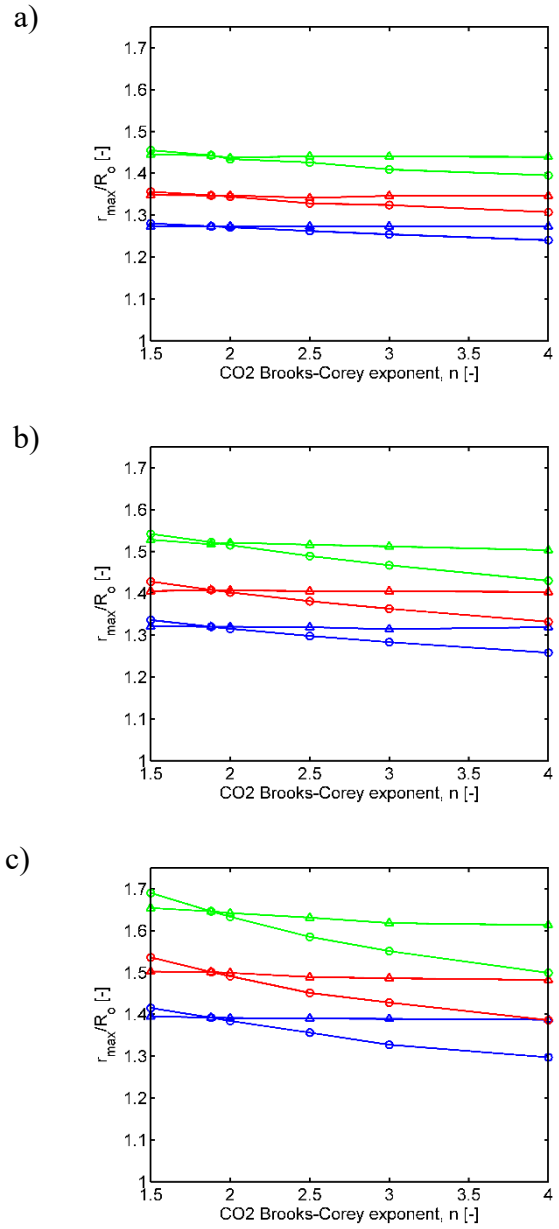


Figure 25. The maximum radial position of mobile CO2, $\tilde{r}_{\max}$, (corresponding to $y_r = 0.05$), as a function of the CO2 Brooks-Corey exponent using the initial condition given by (a) $S_{c,o}/S_c^- = 1.05$, (b) $S_{c,o}/S_c^- = 1.10$, and (c) $S_{c,o}/S_c^- = 1.20$. In all plots, Brooks-Corey model for CO2 relative permeability (circular data points), Brooks-Corey-Hunt model for CO2 relative permeability (triangular data points), Bo $= 0.05$ (blue), Bo $= 0.10$ (red), and Bo $= 0.20$ (green).

3.5. Conclusions

Mitigating the risks associated with sequestered CO2 will prove challenging throughout the lifetime of any reservoir. After the CO2 has become immobile there remains the risk of remobilization over time scales that far exceed most site stewardship considerations. Given these challenges, our analysis provides several important insights into the remobilization and subsequent dynamics of CO2. First, that the susceptibility of CO2 to become remobilized due to a decrease in pore pressure is greatly reduced by storing the CO2 at higher pressure (Figure 11). Usually, such higher initial pressures correspond to deeper injection assuming local pore pressure in the reservoir begins at hydrostatic. Even though there is an increased financial cost to drilling deeper into a formation, the benefits of increased reservoir security due to reduced remobilization scale non-linearly with depth and should be considered in cost-benefit analyses of sequestration projects.

Second, small perturbations in the pressure field will only remobilize a small fraction of CO2, regardless of depth. The local CO2 saturation is predicted to be near the residually trapped limit upon depressurization. As a result, it is crucial that the behavior of the CO2 relative permeability near this limit be well understood and characterized. In most published data sets, too few experimental data points exist near the residual saturation limit to confidently fit a relative permeability model. Nevertheless, the two data sets presented here (from Bennion and Bachu) show better agreement with the Brooks-Corey-Hunt (BCH) model that includes insights from percolation theory than with the Brooks-Corey (BC) model. For most reservoir formations, the best-fit BC model exponent for CO2 is between 2 and 5. If these rocks

display similar trends in relative permeability, then large differences in the predicted dynamics of remobilized CO2 can be expected between the BCH and BC models. More importantly, the BCH model predicts CO2 motion to occur $10^0 – 10^2$ times faster than the BC model in both the vertical and horizontal directions; in many reservoirs this discrepancy may be the difference between several months and several decades. From a reservoir engineering standpoint, this time difference significantly affects the urgency associated with managing a depressurization event.

Finally, we note that the scaling analysis presented in this study assumes that the permeability is isotropic and homogenous throughout the reservoir and that the radial extent of the plume is much larger than the vertical thickness of the domain. As a result, radial flow is predicted to require significantly longer times to occur than the vertical migration of remobilized CO2. Nevertheless, this horizontal spreading of CO2 is important because it increases the risk of leakage through faults or fractures that may be present in the far-field of the caprock. Future work on this theme may also consider anisotropic permeability within the injection zone. Depending on the preferential direction of flow, the time scales for z and r-direction motion may be comparable. In such a system, it would be inappropriate to separate the analysis into a vertical and radial solution. Instead, a full 3-D simulation may be required to predict the migration and equilibration time of remobilized CO2.

This study underlines the potential liabilities of oversimplified analyses of reservoir properties and dynamics, and motivates the need for careful consideration of post-injection dynamics.

CHAPTER 4

MODELING THE DYNAMICS OF MASS TRANSPORT AND CAVITATION WITHIN NANOPOROUS MEDIA CONTAINING MACROSCOPIC INCLUSIONS

4.1. Introduction

As carbon sequestration is being tested in pilot-scale projects throughout the United States, the long term fate and stability of the injected CO2 plume remains difficult to model and predict (DOE, 2012; Rodosta et al., 2014). Of particular importance is the reservoir caprock, which is typically a brine-wetted, low-permeability zone of tight shale or clay located above the injection reservoir. The requirement of this zone is that it provide a physical barrier to the CO2 which is more buoyant than the native brine (Rutqvist and Tsang, 2002; Shukla et al., 2010; Wollenweber et al., 2010). Flow resistance is dependent upon not only the smaller pore sizes and low porosity of the caprock, but also in the capillary barrier between the CO2 domain and the brine-wetted caprock (Burnside and Naylor, 2014; Erendi and Cathles, 2001; Shosa and Cathles, 2001). Without these barriers, injected CO2 would simply migrate back to the earth's surface and reenter the atmosphere. Therefore, it is crucial that the integrity of the caprock and its capillary seals be maintained over long time periods.

Because carbon sequestration involves the injection of pure CO2, brine will spontaneously evaporate into the CO2 when the two liquids share an interface until the concentration of water vapor in the CO2 reaches its saturated value. Near the well head,

this water evaporation can cause dry-out, which manifests itself through the decrease in permeability as pores become constricted by precipitated salts and is an area of ongoing research (Müller, 2010). Although caprock dry-out is a less concerning problem than the direct vertical migration of CO2 through fractures, it may pose additional risks of long term CO2 migration out of the reservoir. In this chapter, we investigate the dynamics of caprock drying by considering an idealized geometry of a nanoporous medium containing larger, macro-sized voids. These voids will have a characteristic length scale of $O(10 - 100\ \mu m)$ and are meant to represent sparse cavities often found within rocks creating a global geometry that is heterogenous, as shown in Figure 26a,c. Using this geometry, we present two different mathematical models to predict the behavior of the pressure field within the porous medium, one using explicit formulations derived separately for the nano and mico porous regions and one using an effective medium formulation to represent the entire domain.

In addition to the mass transport of liquid water, we also consider the dynamics of cavitation occurring within the micro-size voids of the domain and incorporate these physics into both of the models presented. Cavitation can occur when liquid water is placed under tension and is metastable with respect to its vapor phase. As tension in the liquid increases, the liquid water may break (boil) and form water vapor. Although cavitation is typically not considered likely to arise in the subsurface due to the large hydrostatic pressure suppressing the generation of tension, the nanoporous geometry of the caprock coupled with the large stresses generated at the liquid water-dry CO2 interface mean that the possibility of cavitation cannot be excluded.

Finally, we note that the models and methods presented here are not restricted

to use in describing the drying of subsurface caprock. Broadly speaking, the drying of porous media and the dynamics of cavitation are important phenomena in many fields including plant biology, material science, and structural mechanics (Liu et al., 2008; Vincent et al., 2014). In particular, the geometry of our model is well suited to study water transport in xylem tissue of plants, depicted in Figure 26b, in which each xylem cell acts as a large void separated from its cell neighbors by nanoporous pathways called pit membranes.

4.2. Analysis

Unlike the analyses presented in chapters 2 and 3, the focus of this analysis is the mass transport of liquid water within the caprock of a carbon sequestration reservoir. The caprock is a separate geologic domain adjacent to and vertically above the primary injection zone, shown schematically in Figure 26a. At many injection sites, the caprock is made of finely packed particles, such as silts, clays, and shales. As a result of the small pore size, $\mathcal{O}(1\ \mathrm{nm})$, the permeability of caprock is generally 10^2 – 10^7 times lower than the permeability of the injection reservoir (caprock permeability 10^{-15} – 10^{-22} m^2 compared to injection zone permeability 10^{-12} – 10^{-15} m^2). Despite such low permeability, injection zones are usually modeled as being hydraulically connected to the surface as liquid pressures are known to have equilibrated over geologic time scales. Assuming CO2 is sequestered at depths between 1000 and 2000 meters, the hydrostatic liquid brine pressure, $P_b = \rho_b g h$, within caprock would be $10 - 20$ MPa.

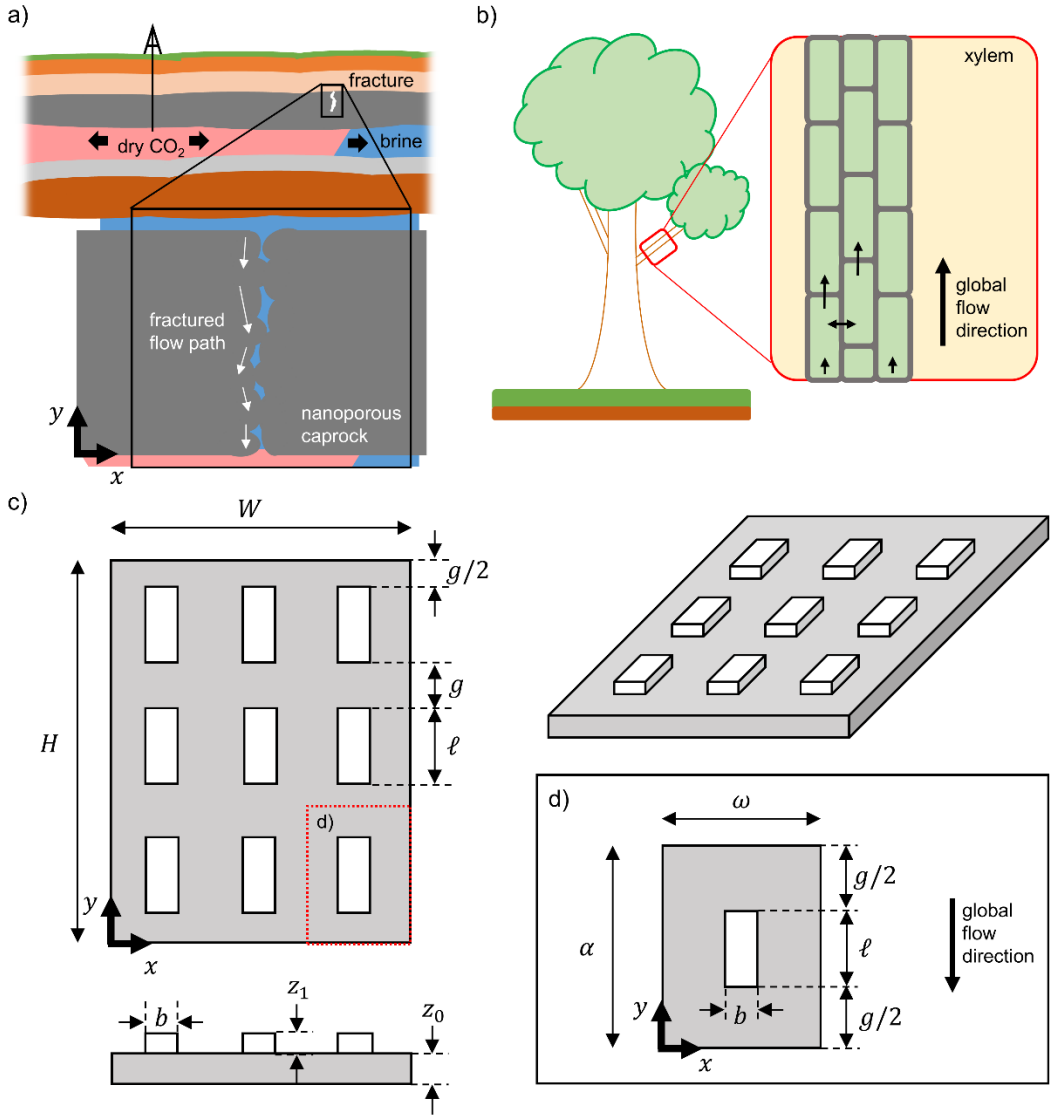


Figure 26. Overview of the nanoporous geometry considered in this analysis. (a) Schematic of a sequestration reservoir. The nanoporous caprock (grey) will contain micro-void inclusions throughout its structure and is assumed to be wetted. In the presence of CO2, drying of the caprock may result in CO2 leaking through fractured pathways within the caprock. (b) Schematic of xylem vessels in a woody plant. Each xylem vessel acts as a conduit through which water is transported by negative pressure. Xylem behaves as a collection of large voids hydraulically connected via nanoporous cell walls. (c) Schematic of the model material used in this analysis. The nanoporous substrate (grey) is hydraulically coupled to an array of larger micro-sized voids of rectangular geometry. The global array of voids can be discretized into repeating units of a single void surrounded by nanoporous media, shown in the (d) inset.

Due to the corrosive, acidic nature of CO2-water mixtures, CO2 is transported and injected dry (pure) at a concentration >99% (DOE, 2012). During CO2 injection, this dry CO2 convects radially outward from the injection well displacing the native brine and, due to buoyancy, is driven upward and into direct contact with the underside of the caprock. Once interfacial contact between CO2 in the reservoir and brine in the caprock exists, water will evaporate into the supercritical CO2 until saturation is reached ($\sim 0.003\ \mathrm{kg_{H20}/kg_{CO2}}$ at 20 MPa and 308 K, but, in general, is a function of temperature and pressure) (Spycher et al., 2003; Spycher and Pruess, 2009). Near the injection well, liquid within caprock is continually exposed to subsaturated CO2 and as a result, mass transport of water from the caprock into the advancing front of CO2 will occur.

While natural caprock will be heterogeneous in permeability and porosity (depicted in Figure 26a), here we will analyze an idealized porous geometry, shown in Figure 26c. This geometry is inspired by recent experimental work by Vincent et. al, included in Appendix 8 (Vincent et al., 2014). It is comprised of a homogenous, isotropic, nanoporous medium (colored in grey) of height H [m], width W [m], and thickness z_o [m]. Note that in this orientation the y-direction is parallel with the gravitational field and the x-z plane is horizontal (perpendicular to gravity). We assume that pressure variations in the fluid are negligible in the z-direction such that the pressure field can be treated as 2-D; a geometry consistent with the presence of a vertical fault or fracture within the caprock. Hydraulically connected to the nanoporous medium are large, rectangular voids of height ℓ [m], width b [m], and thickness z_1 [m], arranged in an ordered array of equal spacing. Inside the pore spaces of this geometry, we will

assume the only phase present is pure water and neglect the presence of salts or other fluids that may be present in a natural caprock.

The dynamics of water mass transport through this geometry will be modeled using two different methods. The first method will involve deriving the mass conservation equations for both the nanoporous substrate and the macro-voids, and coupling the solution of these two domains at their shared boundaries. This method will be referred to as the 'microscale model.' The second method will discretize the global domain into an array of identical, individual elements, depicted in Figure 26d, and derive an effective medium treatment of this individual element that can be reconstructed into an array of elements to capture the dynamics of the original domain. This method will be referred to as the 'effective medium model.'

4.2.1 Governing equations: microscale model

Since we assume that water is the only liquid present in the caprock, its saturation, S_w, must be equal to 1 without cavitation and the conservation of mass within the domain is given by

$$\frac{\partial}{\partial t}[\phi \rho_w] + \boldsymbol{\nabla} \cdot (\rho_w \boldsymbol{u_d}) = 0 \qquad \textbf{Eq. (90)}$$

where ϕ is the porosity of the domain [-], ρ_w is the density of water [kg·m^{-3}], and $\boldsymbol{u_d}$ is the familiar Darcy velocity, defined as

$$\boldsymbol{u_d} = \frac{\kappa}{\mu_w}[\boldsymbol{\nabla}(-P) + \rho_w \boldsymbol{g}]\,. \qquad \textbf{Eq. (91)}$$

Here, we will not assume that the water is incompressible, but instead that it can be elastically deformed according to its adiabatic compressibility, given by

$$B = \rho_w \frac{dP}{d\rho_w} = -\mathcal{v} \frac{dP}{d\mathcal{v}} \qquad \textbf{Eq. (92)}$$

where B is the bulk modulus of water [Pa] ($\sim 2.2\ GPa$) and $\mathcal{v}$ is the molar volume of water [$m^3 \cdot mol^{-1}$]. We note that we are neglecting deformation of the host matrix, since we assume that the elastic modulus of the rock is much larger than the bulk modulus of the water. We will also assume that the caprock height, H, is small, such that hydrostatic pressure differences within the domain are negligible and gravity can be neglected. Using equation 92, the densities in equation 90 can be written in terms of the fluid pressure and after some algebra the conservation of mass can be written as

$$\frac{\partial P}{\partial t} = \frac{1}{\phi} \boldsymbol{\nabla} \cdot \left(\frac{B\kappa}{\mu_w} \boldsymbol{\nabla} P \right) + \frac{\kappa}{\phi \mu_w} (\boldsymbol{\nabla} P \cdot \boldsymbol{\nabla} P) \,. \qquad \textbf{Eq. (93)}$$

where we have assumed that the porosity, ϕ [-], and dynamic viscosity, μ_w [Pa·s], are constant and that total density variations in the water will remain small relative to the equilibrium density ($\Delta\rho_w / \rho_{wo} \ll 1$). As written, equation 93 is the general poroelastic diffusion equation, a nonlinear *PDE* that describes how the pressure field will evolve within a substance. The first term on the right hand side captures the diffusive nature of pressure disturbances within the pore fluid; this would be analogous to thermal diffusion within a homogenous medium. The second term on the right hand side of equation 93 is nonlinear and comes from describing how $dP/d\mathcal{v}$ changes with pressure. Scaling this equation reveals that the nonlinear term on the right hand side can be neglected when $\Delta P \ll B$. To determine if this condition is met, we must consider the pressure generated

at the boundary of the domain. To make an estimate of this pressure, we can assume (though it is not a requirement) that the nanoporous substrate is made of uniform pores of radius r_p [m], the capillary pressure is given by the Young-Laplace equation, written for a cylindrical pore as

$$\Delta P = -\frac{2\sigma \cos\theta_r}{r_p}, \qquad \textbf{Eq. (94)}$$

where σ is the surface tension of the brine-CO2 interface [N·m] and θ_r is the receding contact angle of the meniscus on the walls of the pores. Assuming the contact line is pinned at the caprock-injection zone boundary, this pressure drop must be equal to the pressure difference generated by the liquid brine being in equilibrium with its subsaturated vapor, given by the Kelvin equation

$$\Delta P = \frac{RT}{\mathcal{v}} \ln\left(\frac{p_v}{p_{sat}}\right), \qquad \textbf{Eq. (95)}$$

where $\mathcal{v}$ is the molar volume of liquid water ($\sim 1.805 \times 10^{-5}$ m^3/mol), RT is the thermal energy, and p_v/p_{sat} is the relative humidity of water vapor in the supercritical CO2 (Vincent et al., 2014). If the incoming CO2 is completely dry, then the relative humidity would be zero and the right hand side of equation 95 would approach negative infinity. However, since the injection zone and caprock are initially wetted, incoming supercritical CO2 will quickly absorb water such that the relative humidity will be greater than zero. For relative humidity of 1% ($\Delta P \sim 670$ MPa), the ratio $\Delta P/B$ is approximately equal to 0.3, with higher relative humidity corresponding to smaller values of $\Delta P/B$. We therefore will assume that $\Delta P/B \ll 1$ such that the nonlinear term ($\boldsymbol{\nabla} P \cdot \boldsymbol{\nabla} P$) in equation 93 can be neglected. In a natural porous media with a distribution

of pores, it would be prudent to repeat this scaling with the smallest pore size, as this would correspond to the largest ratio of $\Delta P/B$. Treating the permeability and viscosity as homogeneous and isotropic, equation 93 can now be written as

$$\frac{\partial P}{\partial t} = C\nabla^2 P \qquad \textbf{Eq. (96)}$$

where C is the poroelastic diffusivity [$m^2 \cdot s^{-1}$], defined as

$$C = \frac{\kappa B}{\mu \phi}. \qquad \textbf{Eq. (97)}$$

By assuming the pressure field is two dimensional, equation 96 can be written as

$$\frac{\partial P_i}{\partial t} = C_i \left(\frac{\partial^2 P_i}{\partial x^2} + \frac{\partial^2 P_i}{\partial y^2} \right) \qquad \textbf{Eq. (98)}$$

where the subscript *i* has been introduced to emphasis that this equation is valid in a particular region of constant porosity and permeability. In our model geometry, depicted in Figure 26c, there are two domains to consider: the nanoporous bulk medium, denoted in grey, and the microscale voids, denoted in white. These domains will be noted with subscripts 0 and 1, respectively.

The mass transport of water in each domain will be governed by equation 98. At the interfaces between the nanoporous substrate and the voids, two boundary conditions are required to couple the solutions of each domain. First, the pressure at the boundary must be continuous, implying that

$$P_0(x, y, t) = P_1(x, y, t). \qquad \textbf{Eq. (99)}$$

The second condition requires the flux across the boundary to be continuous. This

condition can be written as

$$\frac{k_0}{\mu}\nabla(-P_0)\cdot\boldsymbol{n}=\frac{k_1}{\mu}\nabla(-P_1)\cdot\boldsymbol{n}\,, \qquad \textbf{Eq. (100)}$$

where $\boldsymbol{n}$ is the vector normal to the surface of the boundary. From equation 98, the characteristic time for pressure relaxation is given by $\tau{\sim}L_{char}^2/C$, where L_{char} is the characteristic distance over which a pressure disturbance must travel. Based on the geometry being analyzed here, the permeability inside the voids, κ_1, is much larger than the permeability of the nanoporous substrate, κ_0. As a result, $C_1 \gg C_2$ and $\tau_1/\tau_0 \ll 1$. The latter scaling result implies that the pressure equilibration inside the voids can be treated as instantaneous with respect to the pressure dynamics occurring within the nanoporous substrate. Therefore, the steady gradients of pressure in each void will be small compared to those in the matrix. Furthermore, when modeling the pressure dynamics of the global domain, we can assume that the pressure inside each void is constant and at pseudo-steady state, where changes in the void pressure can be written as

$$\frac{\mathrm{d}P_1}{dt}=-\frac{B}{V_{v1}}\frac{dV_1}{dt} \qquad \textbf{Eq. (101)}$$

where V_{v1} is the instantaneous volume of water inside the void [m^3] and $\mathrm{d}V_1/dt$ is the instantaneous rate at which the liquid volume is changing inside the void [$m^3{\cdot}s^{-1}$], which can be defined as the water flux integrated over the surface of the void, given by

$$\frac{dV_1}{dt}=-\int \boldsymbol{u}_d\cdot\mathbf{n}\,dA\,. \qquad \textbf{Eq. (102)}$$

To investigate the global transport properties of this porous medium, the outer

boundary conditions of the nanoporous domain will be given by the equations

$$P_0(0 \le x \le W, y = 0) = P_b \,, \qquad \textbf{Eq. (103)}$$

$$P_0(0 \le x \le W, y = H) = 0 \,, \qquad \textbf{Eq. (104)}$$

and

$$\frac{\partial}{\partial x}[P_0(x = 0{,}0 \le y \le H)] = \frac{\partial}{\partial x}[P_0(x = W, 0 \le y \le H)] = 0 \,, \qquad \textbf{Eq. (105)}$$

where P_b is the pressure at the boundary between the caprock and the injection zone and can be determined by the local water vapor pressure in the supercritical CO2 through equation 95. As written, we have subtracted the background hydrostatic pressure in the field, which is given by the hydrostatic pressure at the top of the domain ($P = \rho_w g y(H)$).These boundary conditions impose no flux in the x-direction, which would be physically analogous to the permeability field abruptly changing to zero due to the presence of an impermeable adjacent zone or a periodic array of repeating elements. The boundary conditions at the top and bottom of the domain are set such that at steady state there is a global pressure gradient given by P_b/H.

By introducing the following nondimensional parameters

$$\tilde{x} = x/W \,, \qquad \textbf{Eq. (106)}$$

$$\tilde{y} = y/H \,, \qquad \textbf{Eq. (107)}$$

$$\tilde{t} = tC/H^2 \,, \qquad \textbf{Eq. (108)}$$

$$\tilde{P}_0 = P/P_b \,, \qquad \textbf{Eq. (109)}$$

and

$$\alpha = H/W\,, \qquad \textbf{Eq. (110)}$$

equation 98 can be written in nondimensional form as

$$\frac{\partial \tilde{P}_0}{\partial \tilde{t}} = \alpha^2 \frac{\partial^2 \tilde{P}_0}{\partial \tilde{x}^2} + \frac{\partial^2 \tilde{P}_0}{\partial \tilde{y}^2}\,. \qquad \textbf{Eq. (111)}$$

Similarly, the boundary conditions can be nondimensionalized as

$$\tilde{P}(0 \le \tilde{x} \le 1, \tilde{y} = 0) = 0\,, \qquad \textbf{Eq. (112)}$$

$$\tilde{P}(0 \le \tilde{x} \le 1, \tilde{y} = \alpha) = 1\,, \qquad \textbf{Eq. (113)}$$

and

$$\frac{\partial}{\partial \tilde{x}}[\tilde{P}(\tilde{x} = 0{,}0 \le \tilde{y} \le \alpha)] = \frac{\partial}{\partial \tilde{x}}[\tilde{P}(\tilde{x} = 1{,}0 \le \tilde{y} \le \alpha)] = 0\,. \qquad \textbf{Eq. (114)}$$

These equations describe the pressure field dynamics of the liquid water within the nanoporous domain, subject to a global pressure gradient and accounting for the presence of voids. The pressure within each void is coupled to the solution of equation 111 using nondimensional forms of equations 101 and 102, derived in a similar manner. These equations represent the mass transport equations governing the flow of water through the fully saturated caprock, in the absence of cavitation.

4.2.2 Incorporating cavitation into the microscale model

From recent drying experiments using similar heterogeneous (nanoporous media

coupled to macroscale void inclusions) geometries, it has been shown that cavitation events occur within the larger voids and not in the nanoporous medium (Vincent et al., 2014). The model used here is based on classical nucleation theory (CNT) and has been found to satisfactorily capture the cavitation dynamics observed in these recent experiments, see (Vincent et al., 2014). To begin, the cavitation rate constant, k_{cav} [s^{-1}] for this first order process is given by

$$k_{cav} = V_{v1}\Gamma \qquad \textbf{Eq. (115)}$$

where Γ is the volumetric rate of nucleation [$m^{-3}{\cdot}s^{-1}$], expressed as

$$\Gamma = \Gamma_o exp\left(-\frac{E_{nucl}}{k_B T}\right). \qquad \textbf{Eq. (116)}$$

In this expression, E_{nucl} is the activation energy of nucleation [J], and Γ_o is the attempt frequency [$m^{-3}{\cdot}s^{-1}$]. These are defined as

$$E_{nucl} = \frac{16\pi\sigma^3}{3(P - P_{sat})^2} \qquad \textbf{Eq. (117)}$$

and

$$\Gamma_o = \frac{k_B T}{\hbar\left(\frac{4}{3}\pi R_{crit}^3\right)}, \qquad \textbf{Eq. (118)}$$

respectively. In these expressions, σ is the effective surface tension [$N{\cdot}m^{-1}$], P is the pressure of the liquid inside the void [Pa], P_{sat} is the saturated vapor pressure [Pa], k_B is the Boltzman constant [$J{\cdot}K^{-1}$], T is the liquid temperature [K], $\hbar$ is the Planck constant [J·s], and R_{crit}, defined as

$$R_{crit} = \frac{-2\sigma}{(P - P_{sat})}, \qquad \textbf{Eq. (119)}$$

is the critical radius of nucleation [m]. Substituting these definitions into equation 115 yields

$$k_{cav} = V_w \left(\frac{-3k_B T (P - P_{sat})^3}{32\pi\hbar\sigma^3} \right) exp\left(-\frac{16\pi\sigma^3}{3k_B T (P - P_{sat})^2} \right), \qquad \textbf{Eq. (120)}$$

which is now a single expression for the first order rate process and is a function of the instantaneous void volume and liquid pressure. Using this rate, the probability that no cavitation occurs, $\mathcal{S}$ [-], during an observation time, τ [s], can be expressed as

$$\mathcal{S} = exp(-k_{cav}\tau), \qquad \textbf{Eq. (121)}$$

which implies that the probability cavitation does occur, Π [-], is given by

$$\Pi = 1 - exp(-k_{cav}\tau). \qquad \textbf{Eq. (122)}$$

One important parameter derived from equation 122 is the Π_{50} value, which for a fixed liquid volume and observable time, corresponds to the liquid pressure required to make the probability of cavitation equal to 50% ($\Pi = 0.5$). Note that Π_{50} is a much stronger function of pressure (doubly exponential) than of time. Here, we rely on cavitation experiments performed on a similar porous geometry to determine the value of the effective surface tension required such that Π_{50} corresponds to $P = -23$ MPa and $\tau = 1\ day$. To agree well with experimental results, we find that this model only works when $\sigma \sim 19\ \mathrm{Nm^{-1}}$ and each void in the domain is assigned a surface tension selected from a Gaussian distribution (mean $19\ \mathrm{Nm^{-1}}$, standard deviation $2.5\ \mathrm{Nm^{-1}}$).

This effective surface tension is then used in the numerical simulation.

To include cavitation dynamics in the microscale model, the governing equation 111 for the pressure field is again solved numerically. First, the pressure is updated in the entire domain at each time step using equation 111. Given the instantaneous pressure and liquid volume, the cavitation probability for each void is calculated using equations 120 and 122 assuming that the observable time, τ, is equal to the time step of the simulation ($\tau = \Delta t$). A random number is then generated and if this number is less than Π, then we assume that cavitation has occurred and the liquid pressure inside the void is set to zero ($\sim P_{sat}$), as it will be in equilibrium with its vapor. Since the surrounding porous media is being dried and contains liquid at negative pressure, water will empty from the void. When there is no liquid water remaining in the void, the pressure field inside the void is updated as though it were nanoporous media whose behavior is indistinguishable from the immediate surroundings and is governed by equation 111. This is consistent with the geometry used in lab experiment, which possesses a nanoporous substrate everywhere in the domain, including beneath the voids (though may not exist in all rock applications).

4.2.3 Governing equations: effective medium model

An alternative approach to the microscale model is to consider an effective medium, or lumped, model. Effective medium techniques are employed in a wide range of problems and were most famously used by Maxwell to describe dielectric properties of heterogeneous materials, and the conductance of insulating bulk material containing

sparsely distributed, highly conductive particles (Aspnes, 1981; Levy and Stroud, 1997; Maxwell Garnett, 1904). The approach involves solving Laplace's equation in both phases and so it applies as well to porous media flows as well as thermal conduction and mass diffusion. The general procedure is to consider and solve the global conservative equations (often referred to as the far-field or outer solution) using effective properties in the transport equations, then consider a local solution of the bulk medium surrounding one heterogeneous region or particle (inner solution), then equate the inner and outer solutions to derive the effective transport of the property being modeled. In this case, we are interested in the global fluid conductance (permeability) as well as the global fluid capacitance. One important assumption when building an effective medium model is that the field lines of the property of interest should approach its unperturbed behavior far away from the heterogeneous inclusions. Similarly, this technique requires that the inclusions within the medium behave independently (no particle-particle interactions). These conditions are usually met when the volume fraction of particles is low and as a result, the majority of effective medium models have been derived for this volume fraction regime. However, we note that diluteness of heterogeneities is not a requirement to derive an effective medium treatment; geometries containing periodic unit cells can also be analyzed with this technique even when the heterogeneities are large, see (Brenner, 1980)

In the geometry considered here, the global domain can be conveniently discretized into the base unit depicted in Figure 26d, consisting of a single void centered symmetrically in the x and y directions and surrounded by nanoporous media. Again, the global pressure gradient is only in the y-direction, with no flux conditions along the

left and right boundaries, as this is equivalent periodicity with symmetry within the unit cell. If the void were not present, the solution to this flow would consist of parallel, vertical streamlines beginning at $y = \alpha$ and ending at $y = 0$, with the total flux determined by the magnitude of the pressure gradient and permeability of the nanoporous medium. Because of this symmetry (no lateral flux between elements), we can first consider the dynamics within a single unit cell and recognize that the global domain can be described by the linear combination of unit cells.

To nondimensionalize this analysis, we will define several parameters relating the void size to the size of the surrounding porous medium. The first parameter, β [-], is the ratio of the void width to the width of the base unit, given by

$$\beta = \frac{b}{\omega}. \qquad \textbf{Eq. (123)}$$

Since the permeability of the voids is much larger than the permeability of the nanoporous substrate, fluid flow will preferentially pass through the void. Even though the global pressure gradient is purely in the y-direction, the flow field within the material will be 2-dimensional, as the streamlines will arc toward and away from the void at the ends of the domain. The value of β gives some indication of the curvature of the streamlines required for flow to pass through the void. Larger values of β imply that the void occupies a greater fraction of the domain in the x-direction and as a result, the streamlines of the flow need not deviate greatly from a vertical orientation.

The second parameter, λ [-], is the ratio of the void length to the total length of the base unit, given by

$$\lambda = \frac{\ell}{\alpha} = \frac{\ell}{g + \ell}. \qquad \textbf{Eq. (124)}$$

This ratio is helpful in estimating the difference in flow resistance between a vertical streamline passing through the void and a vertical streamline passing only through the nanoporous medium. Larger values of λ imply that the flow resistance is reduced for fluid passing through the void compared to fluid passing only through the nanoporous substrate, since fluid passing through the void must only travel through a distance of $O(g)$ through the more restrictive nanopores. Finally, the ratio of void thickness, z_1 [m], to substrate thickness, z_0 [m], will be defined as ζ [-],

$$\zeta = \frac{z_1}{z_0}. \qquad \textbf{Eq. (125)}$$

In solving the 2-D flow field for this base unit domain containing a single void of arbitrary size, the most important geometric ratio to define is the area ratio of the void relative to the entire nanoporous substrate. Using the definitions of β and λ, this area ratio , η [-], can be written as

$$\eta = \beta\lambda\,, \qquad \textbf{Eq. (126)}$$

see Appendix 7 for a general derivation of equation 126. Using this base geometry, we can derive the pressure dynamics within the domain using an effective medium with a single fluid pressure, capacitance, and permeability. This equation describing the global pressure dynamics can be written as

$$\frac{\partial}{\partial t}\left[\left(1 + \frac{\eta\zeta}{\phi}\right)\langle P\rangle\right] = \frac{Bk_\epsilon}{\phi\mu}\nabla^2\langle P\rangle\,. \qquad \textbf{Eq. (127)}$$

where $\langle P \rangle$ is the volume averaged pressure of the liquid within the unit cell [Pa], and k_ϵ is the effective permeability of this geometry [m^2]. The term $(1 + \eta\zeta/\phi)$ is the effective capacitance of the domain, as it includes all the liquid in the nanoporous substrate as well as the liquid inside the voids. The most important term in this equation is k_ϵ because its role is to account for the change in permeability of the domain resulting from the presence of the voids. Depending on the geometry and spatial arrangement of the voids with respect to the global fluid motion, there are several models that could be used to provide a functional form of k_ϵ. Given that we consider here a rectangular void whose long axis is aligned parallel with the global pressure gradient, we will present two possible forms of k_ϵ to compare to the microscale model.

The first model is a 2-D Maxwell effective medium model that has been adapted for elliptic inclusions aligned with the flow, which we will use to approximate the rectangular voids, (see Appendix 9 for derivation and (Zimmerman, 1996)) and can be written as

$$k_{\epsilon,e} = k_o \left[\frac{1 - G\eta}{1 + G\eta}\right] \qquad \textbf{Eq. (128)}$$

where k_o is the permeability of the nanoporous substrate [m^2], η is the area fraction of the voids in the domain (as defined in equation 126) [-], and G is a factor incorporating the geometry of the voids and the contrast in permeability between the nanoporous substrate and the voids. G is defined as

$$G = \frac{(1 + \mathcal{R}_e)\left(1 - \frac{k_1}{k_o}\right)}{2\left(1 + \mathcal{R}_e \frac{k_1}{k_o}\right)} \qquad \textbf{Eq. (129)}$$

where k_1 is the permeability of the voids [m^2], and $\mathcal{R}_e$ is the aspect ratio of the elliptical inclusions (minor axis:major axis), which can be defined for the rectangular voids here as

$$\mathcal{R}_e = \frac{b}{\ell} = \frac{\omega\beta}{\alpha\lambda}. \qquad \textbf{Eq. (130)}$$

Equation 129 can be simplified further in cases when $k_1 \gg k_o$, as we assume here; and can be written as

$$G = \frac{-(1+\mathcal{R}_e)}{2\mathcal{R}_e}. \qquad \textbf{Eq. (131)}$$

We also note that when the inclusions are circular, given by an aspect ratio equal to 1 ($\mathcal{R}_e = 1$), and when $k_1 \gg k_o$, G becomes equal to -1 and equation 128 becomes

$$k_{\epsilon,e} = k_o\left[1 + \frac{2\eta}{1-\eta}\right], \qquad \textbf{Eq. (132)}$$

which is the more common 2D Maxwell equation for effective permeability (Choy, 1999). In this analysis, equations 128, 130, and 131 will be used to define the Maxwell model.

The second model of effective permeability is a simpler, parallel path description of the fluid flow through the base domain. In this model, flow is treated as one dimensional in the y-direction and as a result, the fraction of fluid flowing through the void is only a function of the facial area fraction that the void occupies in the domain (given by β). Fluid in the outer regions of the domain travel only through the nanoporous substrate. Fluid in the inner region whose vertical path intersects with the void, travels through the nanoporous substrate a distance g and the void a distance ℓ.

The effective permeability of the entire base domain is the inverse of the average of the resistances of these two possible flow paths, weighted by the facial area of the void. Using this approach, the parallel path effective permeability can be written in as

$$k_{\epsilon,p} = k_0(1-\beta) + \frac{k_0 k_1 \beta}{(k_1(1-\lambda) + k_0\lambda)}. \qquad \textbf{Eq. (133)}$$

This function can also be simplified when $k_1 \gg k_o$, allowing $k_{\epsilon,p}$ to be written as

$$k_{\epsilon,p} = k_o\left[(1-\beta) + \frac{\beta}{(1-\lambda)}\right], \qquad \textbf{Eq. (134)}$$

which is the form of the model that will be used here, see Appendix 10 for derivation.

Regardless of the choice of effective permeability, in the absence of cavitation and void emptying the effective capacitance of the domain remains approximately constant with respect to time and, as a result, equation 127 can be written as

$$\frac{\partial\langle P\rangle}{\partial t} = C_\epsilon \nabla^2\langle P\rangle \qquad \textbf{Eq. (135)}$$

where all effects of the inclusions are lumped into an effective poroelastic diffusivity, given by

$$C_\epsilon = \frac{Bk_\epsilon}{\phi\mu} * \left[1 + \frac{\eta\zeta}{\phi}\right]^{-1}. \qquad \textbf{Eq. (136)}$$

4.2.4 Incorporating cavitation into the effective medium model

Equations 135 and 136, along with a suitable function to define k_ϵ, constitute an effective medium model that can be used to predict the global pressure dynamics of the

pore fluid and the mass flux of liquid through the entire domain. However, in order to include cavitation dynamics in this model, several modifications to these equations must be made. Before addressing these changes, we will first introduce several new parameters required to describe the fill-state of the voids within a region. Let n_F, n_A, and n_E be the number density [#·m^{-2}] of uncavitated voids filled with water, cavitated voids that are in the process of emptying (active), and cavitated voids that are empty and no longer participate in the transport dynamics of liquid through the global domain, respectively. Since these three parameters encompass all of the possible fill-states of the voids, the total void density within a region, n_T [#·m^{-2}], must be given by

$$n_T = n_F + n_A + n_E \,. \qquad \textbf{Eq. (137)}$$

Using these definitions of number densities for the three possible void fill states (full, emptying, empty), the effective medium equation governing the pressure dynamics in the global domain can be written as

$$\frac{\partial}{\partial t}\left[\left(1 + (n_F + n_A)\frac{V_w}{\phi z_0}\right)\frac{\langle P\rangle}{B}\right] - \boldsymbol{\nabla}\cdot\left[\frac{k_\epsilon}{\phi\mu}\boldsymbol{\nabla}\langle P\rangle\right] = -\frac{n_A h_p \langle P\rangle}{\phi A z_0} \qquad \textbf{Eq. (138)}$$

where A is the area of the global domain over which the pressure is being averaged [m^2], given by and h_p is a mass transfer coefficient [m^3·Pa^{-1}·s^{-1}] whose value depends on the local fill state of neighboring voids and is defined as

$$h_p = \frac{2\pi k_\epsilon z_0}{\mu \ln\left(\frac{R_s}{R_v}\right)} \qquad \textbf{Eq. (139)}$$

where R_v is the average radius of a single void, given by

$$R_v = \sqrt{b\ell/\pi}\,, \qquad \textbf{Eq. (140)}$$

and

$$R_s = min\left[n_A^{-1/2}, L_{wall}\right]. \qquad \textbf{Eq. (141)}$$

In equation 141, L_{wall} is the distance between the local domain being averaged and the nearest global boundary allowing fluid to escape the domain [m] and $n_A^{-1/2}$ is the average distance between two emptying voids within the local domain [m]. Here, the mass transfer coefficient (equation 139) has been derived by assuming that a single, circular void has cavitated and the pressure field around the void is in pseudo steady state. From this, the pressure field around the void is assumed to decrease logarithmically from the void and the pressure disturbance is assumed to be zero at either the global boundary or the neighboring active voids.

Equation 127 and equation 138 describe the dynamics of the pressure field using an effective medium model, in the absence of cavitation and incorporating cavitation, respectively. Although these equations are similar in their derivation, there are several important differences between the two. First, the capacitance on the left hand side of equation 127 is a constant when cavitation is not considered, since the volume of water within a given region is treated as approximately constant. By including cavitation, in 138, voids that cavitate or empty contain less liquid volume and therefore the total capacitance of the local region depends on the number of full and empty voids. Second, the permeability of the voids remains much larger than the nanoporous permeability so long as the voids contain liquid water. After cavitation and emptying, the voids no longer transport water efficiently, and the effective permeability in the neighborhood of

the void returns to k_0. As a result, when including cavitation in the model, the effective permeability of the global domain is spatially and temporally dependent (the second term in 138). We will limit our cavitation dynamics analysis to the use of the modified Maxwell effective medium model for elliptic inclusions and thus the equation for $k_{\epsilon,e}$ (eq. 128) must be modified such that

$$k_{\epsilon,e} = k_o \left[\frac{1 - G\left(\frac{n_F + n_A}{n_T}\right)\eta}{1 + G\left(\frac{n_F + n_A}{n_T}\right)\eta} \right] \qquad \textbf{Eq. (142)}$$

where η is the total area ratio of voids to neighboring substrate, regardless of the fill state of the voids, and G is defined in equation 131. Finally, the source term on the right hand side of equation 138 is nonexistent in the absence of cavitation. This source term is important because upon cavitation, the liquid pressure inside the void is assumed to be zero because it equilibrates with its vapor and the vapor pressure at saturation is approximately zero. As a result, the emptying of voids within the domain act as a local source of fluid and a source of high pressure (relative to the negative pressures required to cause cavitation).

The dynamics of cavitation and void emptying are included in this model through the use of additional ODEs coupled to equations 137 and 138. First, the rate at which the number density of empty voids grows within a local region is given by

$$\frac{dn_E}{dt} = -n_A \frac{h_p \langle P \rangle}{V_w}. \qquad \textbf{Eq. (143)}$$

This equation treats all active voids within a given unit cell as having a volume, V_w [m^3], that is emptying into the local nanoporous media with a mass transfer coefficient equal

to h_p and a driving force equal to $\langle P \rangle$. Second, the rate at which full voids cavitate is given by

$$\frac{dn_F}{dt} = -n_F \cdot k'_{cav} \qquad \textbf{Eq. (144)}$$

where k'_{cav} is defined as

$$k'_{cav} = (V_w n_F A)\left(\frac{-3k_B T(\langle P \rangle - P_{sat})^3}{32\pi\hbar\sigma^3}\right) exp\left(\frac{-16\pi\sigma^3}{3k_B T(\langle P \rangle - P_{sat})^2}\right), \qquad \textbf{Eq. (145)}$$

similar to equation 120. Finally, these equations are closed by recognizing that n_T is constant and therefore, equation 137 can be differentiated with respect to time to yield

$$0 = \frac{dn_F}{dt} + \frac{dn_A}{dt} + \frac{dn_E}{dt}, \qquad \textbf{Eq. (146)}$$

which relates the time rate of change of emptying voids, dn_A/dt, to equations 143 and 144. Equations 137 – 146 define the effective medium model required to include the dynamics of cavitation and will recover equations 135 and 136, provided that $k_\epsilon = k_{\epsilon,e}$, $k'_{cav} = 0$, $n_A = n_E = 0$, and $n_F = 1$.

4.3. Discussion and results

In this work, we investigate several important transport characteristics of the model domain as well as the pressure dynamics associated with liquid draining from the porous media both with and without cavitation. These properties and dynamics are predicted using the microscale and effective medium models outlined and described in section 4.2 (code provided in Appendix 11 and Appendix 12). Unless otherwise stated,

the properties listed in Table 7 were used as inputs to both models.

Properties of the nanoporous substrate	
porosity of nanoporous substrate, ϕ [-]	0.5
permeability of the nanoporous substrate, k_0 [m^2]	1.9 x 10^{-20}
depth of nanoporous substrate, z_0 [μm]	5.0
width of global domain, W [μm]	2000.0
length of global domain, H [μm]	2000.0
width of base domain, ω [μm]	200.0
length of base domain, α [μm]	200.0
Properties of the liquid	
viscosity, μ [Pa·s]	1 x 10^{-3}
effective surface tension, σ [N·m^{-1}]	0.019
Properties of the voids	
void depth, z_1 [μm]	20
total number of voids in domain and array arrangement	100 [10 x 10]
nondimensional void width, β [-]	0.5
void length:length of base domain, λ [-]	0.5
void area fraction, η [-]	0.25

Table 7. The parameter values for the nanoporous substrate, microscale voids, and water used in this analysis, unless otherwise noted.

4.3.1 Effective permeability

The first and most important characteristic investigated was the effective permeability of the global domain, which was found in the absence of cavitation. The effective permeability is crucial because it affects both the transient pressure dynamics and the steady state flux through the domain. Figure 27 presents the effective permeability found from simulation using the microscale and effective medium models. These numerical simulations were performed for the range of $0.1 \leq \beta \leq 0.9$ and $0.25 \leq \lambda \leq 0.9$, which corresponds to a void area fraction range of $0.025 \leq \eta \leq 0.81$, where, again, β is the ratio of the void width to the total width of the unit cell, λ is the

ratio of the void length to the total length of the unit cell, and the Maxwell effective model is expected to apply if η is small. Comparing the effective medium results to those from the microscale model, we see that the Maxwell elliptic void effective medium model (Figure 27, blue curves) is generally in better agreement with the microscale model than the parallel path model (Figure 27, red curves). For large values of β, the elliptic model tends to over-predict the effective permeability while the parallel path model consistently under-predicts the effective permeability for all β. This latter relationship is expected because the parallel path model is incapable of including the 2-D nature of the streamlines near each void that would tend to increase the volume of fluid passing through the high permeability center of the domain. More importantly, the parallel path model is able to better capture the curvature in the k_ϵ/k_0 data points at larger values of λ. This may be explained by two effects: first, at large values of λ the voids nearly span the entire domain, implying that the flow resistance from the nanoporous media is nearly short-circuited by the highly conductive voids which nearly span the entire domain and the conductance approaches infinity. The parallel path model is better adapted to handle this configuration provided some small gap distance, $g/2$, remains. Second, the Maxwell elliptic model is derived assuming voids are sparsely distributed and that the disturbance to the pressure field from neighboring voids does not overlap; a condition not well met as $\lambda \to 1$.

Given these differences between the Maxwell and parallel path models, one may propose a blended equation which combines these two effective medium treatments into a single expression such as

$$k_{\epsilon,tot} = (1-\lambda)k_{\epsilon,e} + \lambda k_{\epsilon,p}\ , \qquad \textbf{Eq. (147)}$$

which would weigh the Maxwell model more at smaller values of λ and the parallel path model more at larger values of λ. This blended model is compared to the results from the microscale model in Figure 28. While this blended model does a better job capturing the curvature in the data of the microscale model, it does not reproduce the effective permeability values found by the microscale model, and further modification to equation 147, such as additional prefactors, would be necessary to best match the microscale model.

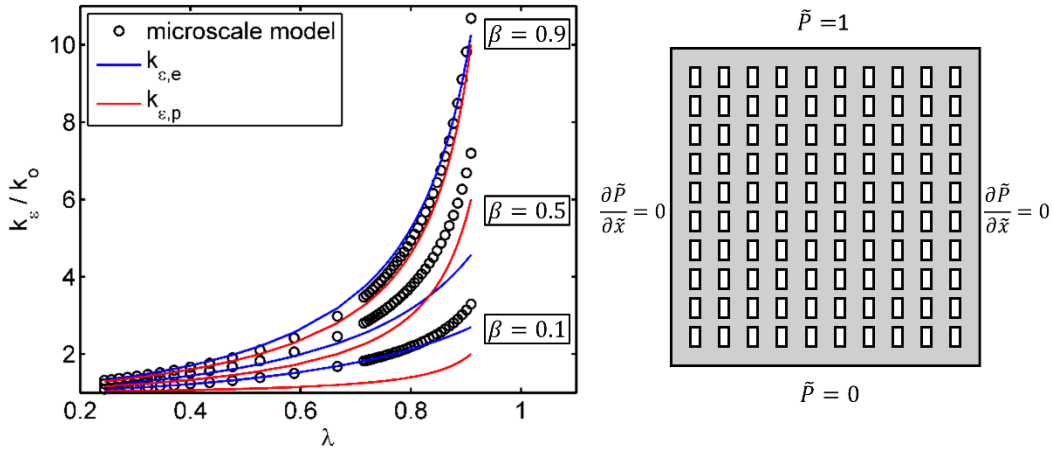


Figure 27. (Left) Effective permeability of the global domain predicted by the microscale model, found by solving equation 111 and nondimensional forms of equations 101 and 102 (black circles), Maxwell's elliptic effective medium model described by equation 128 (blue), and the parallel path model described by equation 134 (red). (Right) Schematic of geometry used in the microscale model and the corresponding boundary conditions applied to the global domain.

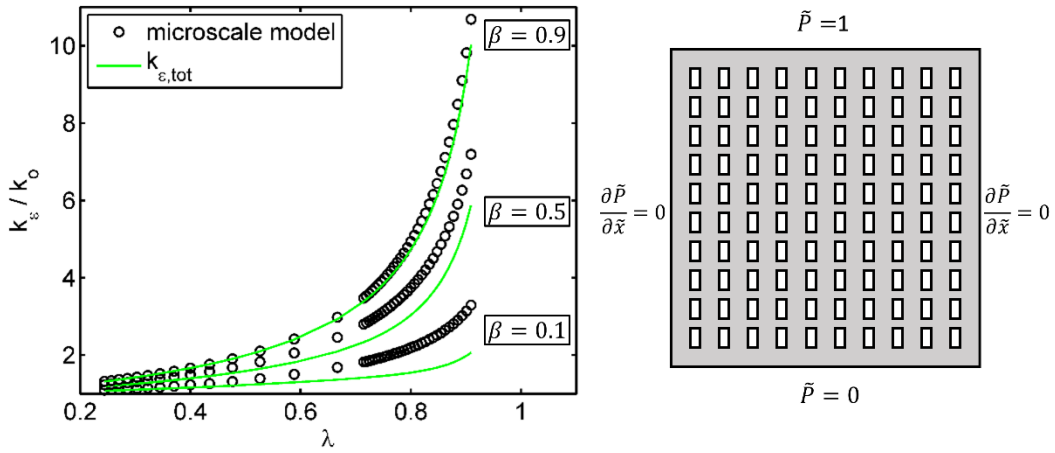


Figure 28. (Left) Effective permeability of the global domain predicted by the microscale model, found by solving equation 111 and nondimensional forms of equations 101 and 102 (black circles) and the blended effective medium model defined in equation 147 (green). (Right) Schematic of geometry used in the microscale model and the corresponding boundary conditions applied to the global domain.

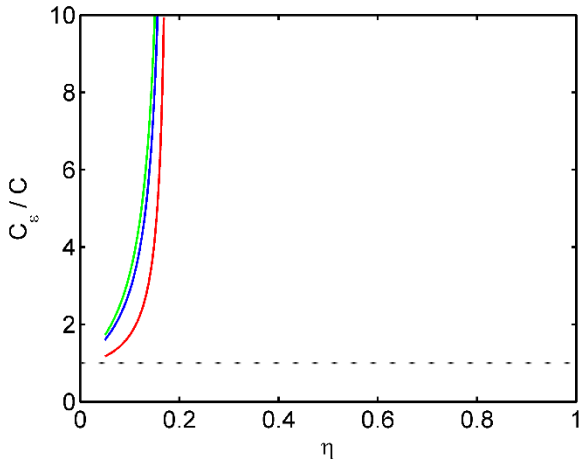
4.3.2 Effective capacitance

Closely related to the effective permeability, the effective capacitance of the global domain influences the characteristic time for transient disturbances in the pressure field to decay. From section 4.2, $\tau \sim L_{char}^2 / C$ in the absence of voids, where C is defined in equation 97. This characteristic time is modified when modeling the system as an effective medium and can be written as $\tau_\epsilon \sim L_{char}^2 / C_\epsilon$, where C_ϵ is defined in equation 136. From these definitions of poroelastic diffusivity, the ratio of characteristic relaxation times in the absence of voids to that with voids can be written as

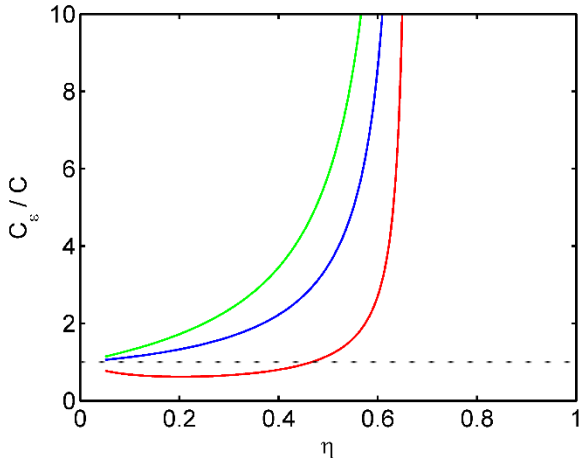
$$\frac{\tau}{\tau_\epsilon} = \frac{C_\epsilon}{C} = \left(\frac{1 + \eta\left(\frac{1+\mathcal{R}_e}{2\mathcal{R}_e}\right)}{1 - \eta\left(\frac{1+\mathcal{R}_e}{2\mathcal{R}_e}\right)} \right) \left[1 + \frac{\eta\zeta}{\phi} \right]^{-1}, \qquad \textbf{Eq. (148)}$$

where, again, η is the area fraction occupied by the voids, ζ is the ratio of void height to nanoporous domain height, and $\mathcal{R}_e$ is the aspect ratio of the void. The relationship expressed in equation 148 is important because it allows one to predict how much faster or slower mass transport may occur if voids are present based on the geometry of the void. While voids occupying a larger area fraction of the domain tend to increase the rate of mass transport, if the voids add significant liquid volume to the global domain, then this added capacitance will tend to slow the rate of mass transport. To demonstrate, τ/τ_ϵ has been plotted for several values of η, ζ, and $\mathcal{R}_e$ in Figure 29. In each plot, the ratio $\tau/\tau_\epsilon = 1$ is denoted by the black dashed line. Interestingly, for larger values of ζ and $\mathcal{R}_e$, mass transport is slowed by the presence of voids because the increase in permeability is not outweighed by the additional capacitance of the liquid added to the system, while in most cases, the presence of voids is predicted to reduce the characteristic relaxation time of the system by a factor of 2 – 10. This increase in diffusivity is significant with respect to caprock drying as it suggests that the presence of undetected voids or fractures in the rock will likely increase the rate of drying making the caprock more susceptible to failure. In all of these plots, the voids are modeled as being elliptic, where $\mathcal{R}_e$ is the aspect ratio (minor axis:major axis), and the major axis of the ellipse is aligned with the flow. In Figure 29c, $\mathcal{R}_e = 1$, which corresponds to the voids being modeled as circular inclusions in the domain.

a)



b)



c)

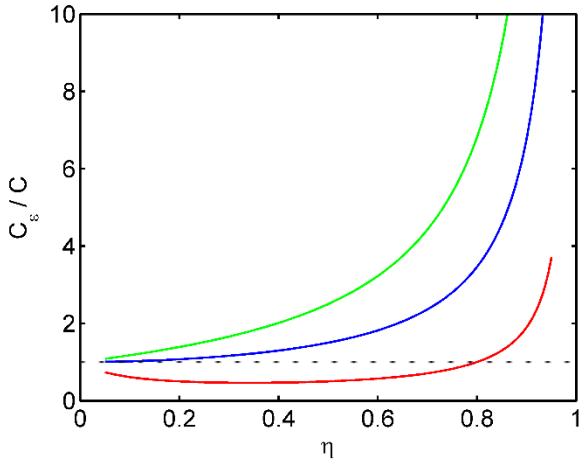


Figure 29. Ratio of effective poroelastic diffusivity in the presence of voids to that without voids, C_ϵ/C, as a function of η for $\mathcal{R}_e = 0.1$ (a), 0.5 (b), and 1 (c), as defined in equation 148. In all plots $\zeta = 0.2$ (green), 1 (blue), and 5 (red). The black dashed line corresponds to $C_\epsilon/C = 1$. Note that C_ϵ is defined in equation 136 and C is defined in equation 97.

4.3.3 Drying dynamics with and without cavitation

Finally, we investigate the dynamic pressure response of the fluid in the global domain as a result of a step decrease in pressure at one boundary while holding the opposite boundary at fixed pressure. Consistent with the boundary conditions defined in equations 103, 104, and 105 as well as the parameters defined in Table 7, the microscale and elliptic void effective medium models were first used to simulate the transient pressure response to the condition $P_b = -10\ MPa$. Under these conditions, no cavitation is predicted to occur because the pressure inside the domain is far above the value of $\mathcal{P}_{50}$, which is approximately $-23\ MPa$. The pressure field predicted by each model is presented in Figure 30 for three distinct times during the pressure relaxation. To display the pressure along the y-axis of the microscale model in a single function, at each y-location the pressure field was averaged along the x-direction and this average pressure is presented in the figure. Throughout the transient response, the microscale model predicts that pressure dynamics occur faster, observable in Figure 30 as the black line falling on average below the red line, and reaches steady state in approximately 15 seconds, compared to the effective medium model which reaches steady state in approximately 17 seconds (we treated the solution as being in steady state when the sum of the residuals of the numerical solution were less than 1 x 10^{-6}). Note that the step-like shape of the microscale model is the result of the pressure field being uniform within the voids and exhibiting steep gradients between neighboring voids, while the effective medium model smooths the pressure field within the region of each void. This difference in time to reach steady-state can be explained by the higher permeability predicted by the microscale model compared to the Maxwell effective medium model, as discussed

in section 4.3.1 and depicted in Figure 27. Despite the small difference in relaxation time, these models show strong agreement with one another in this transient pressure test.

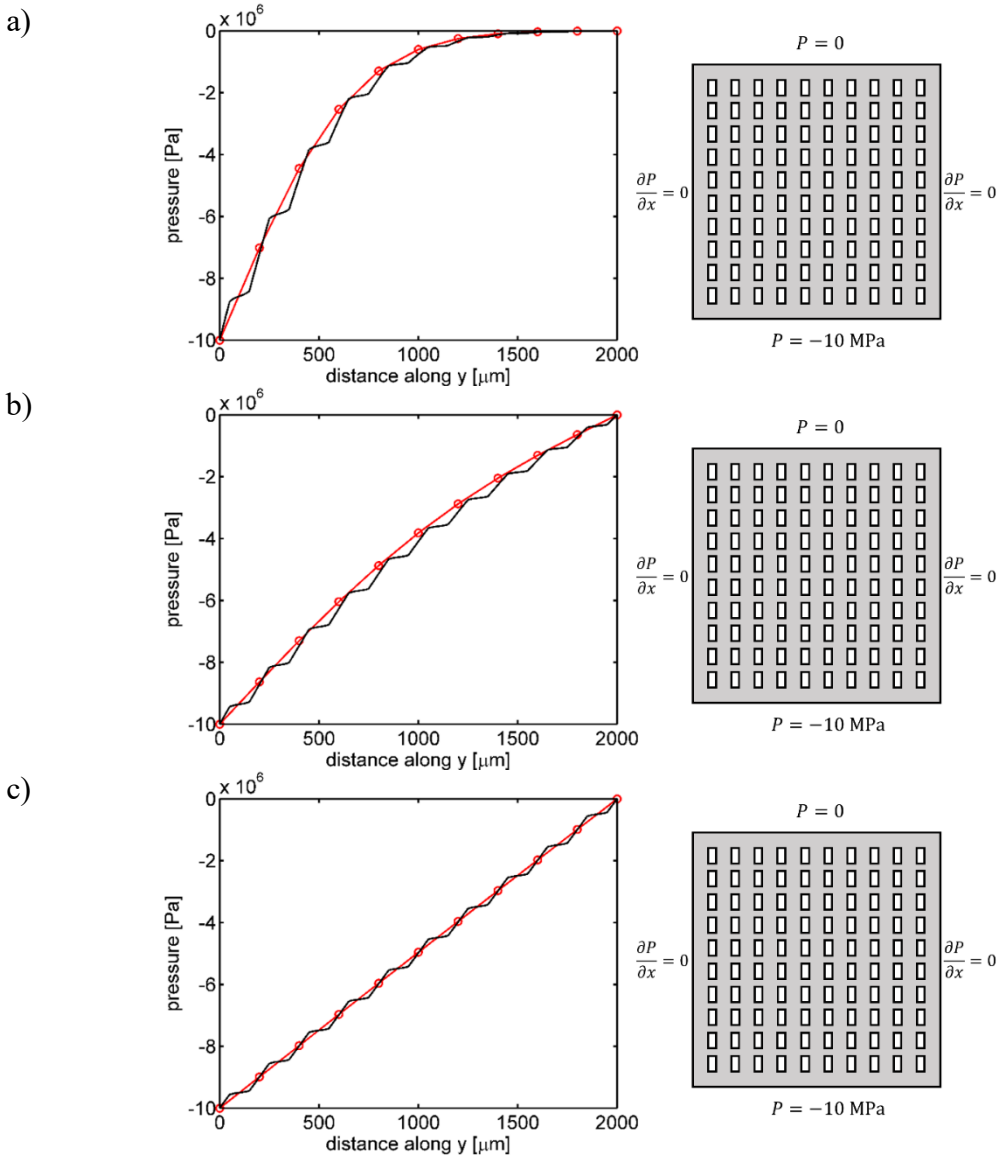


Figure 30. (Left) Transient pressure response in the absence of cavitation. Pressure profile in the y-direction across the global domain at times (a) 1 s, (b) 5 s, and (c) 15 s, after pressure at bottom boundary (y=0) jumps from 0 to -10 MPa, using the microscale model, averaged in the x-direction (black), and Maxwell's elliptic effective medium model (red). (Right) Schematic of geometry used in the microscale model and the corresponding boundary conditions applied to the global domain.

This transient pressure response test was repeated using $P_b = -45\ MPa$. With this boundary condition, cavitation of the voids is predicted to occur, beginning with the voids nearest the drying boundary ($y = 0$). Similar to the first transient test (Figure 30), the pressure field predicted by each model is presented in Figure 31 for three distinct times during the pressure relaxation. In these plots, the green line corresponds to the location of $\mathcal{P}_{50}$ in the steady state pressure profile. Voids to the left of this line (closer to the drying boundary) are expected to cavitate, while voids to the right of this line (further from the drying boundary) are expected to remain full.

Three important differences are observed in this case with cavitation compared to the previous case, which was without cavitation events. First, each model predicts a much longer time required to reach steady state (~100 seconds in both models, Figure 31c, compared to ~15 seconds in the previous test, Figure 30c). This is due to the fact that additional time is required for cavitated voids to completely empty before the pressure field around these voids can equilibrate. This longer transient is also related to the second notable difference: cavitated voids no longer aide in increasing the permeability of the local domain. This implies that, as the first row of voids nearest to the drying boundary cavitate and empty, the subsequent pressure relaxation is retarded due to the reduced permeability of this region. As a result, the second row of voids take longer to reach a reduced pressure such that they cavitate and empty, and so on. This effect can be observed in Figure 31c, where the pressure field of the microscale model is a straight line to the left of the $\mathcal{P}_{50}$ marker and returns to a stair-like function to the right of the $\mathcal{P}_{50}$ marker because the permeability in the region closer to the boundary is the same everywhere since all the voids in this region have emptied. This difference in

permeability also explains the change in slope observed in the pressure field at the $\mathcal{P}_{50}$ value in Figure 31c as the flux through the domain is constant at this time.

Third, unlike the previous case, in this simulation the Maxwell effective medium model predicts pressure equilibration occurs faster than the microscale model. Although it was demonstrated that the microscale model predicts a higher effective permeability when the domain contains voids filled with liquid (Figure 27), the emptying time of voids is longer in the microscale model than in the effective medium model. To quantify this difference, the fraction of empty voids in the first four rows nearest to the drying boundary of the microscale model are plotted as a function of time in Figure 32. Included in this figure is also the values of n_E, which is the number of empty voids per unit area, found from the effective medium model at the same y-location for each row. This data shows that the row of voids closest to the drying boundary actually empty sooner in the microscale model than in the effective medium model, but in all subsequent rows, the time required to empty all voids lag behind the effective medium model with this time lag increasing with each row. This lag could be caused by several factors. First, in the microscale model, any cavitation event immediately causes the pressure field inside the void to rise to 0 and remain 0 until the void is empty. In the effective medium model, voids that are actively emptying within a region contribute to a local increase in pressure captured by the right hand side of equation 138, but this term may not capture the impact of emptying on the pressure field accurately: the time required for each void to empty is estimated in the effective medium model using the mass transfer coefficient, h, which assumes a pseudo steady-state, radially symmetric pressure profile around each emptying void; this condition is not truly met in this

situation as global transport is preferentially directed toward the drying boundary of the domain.

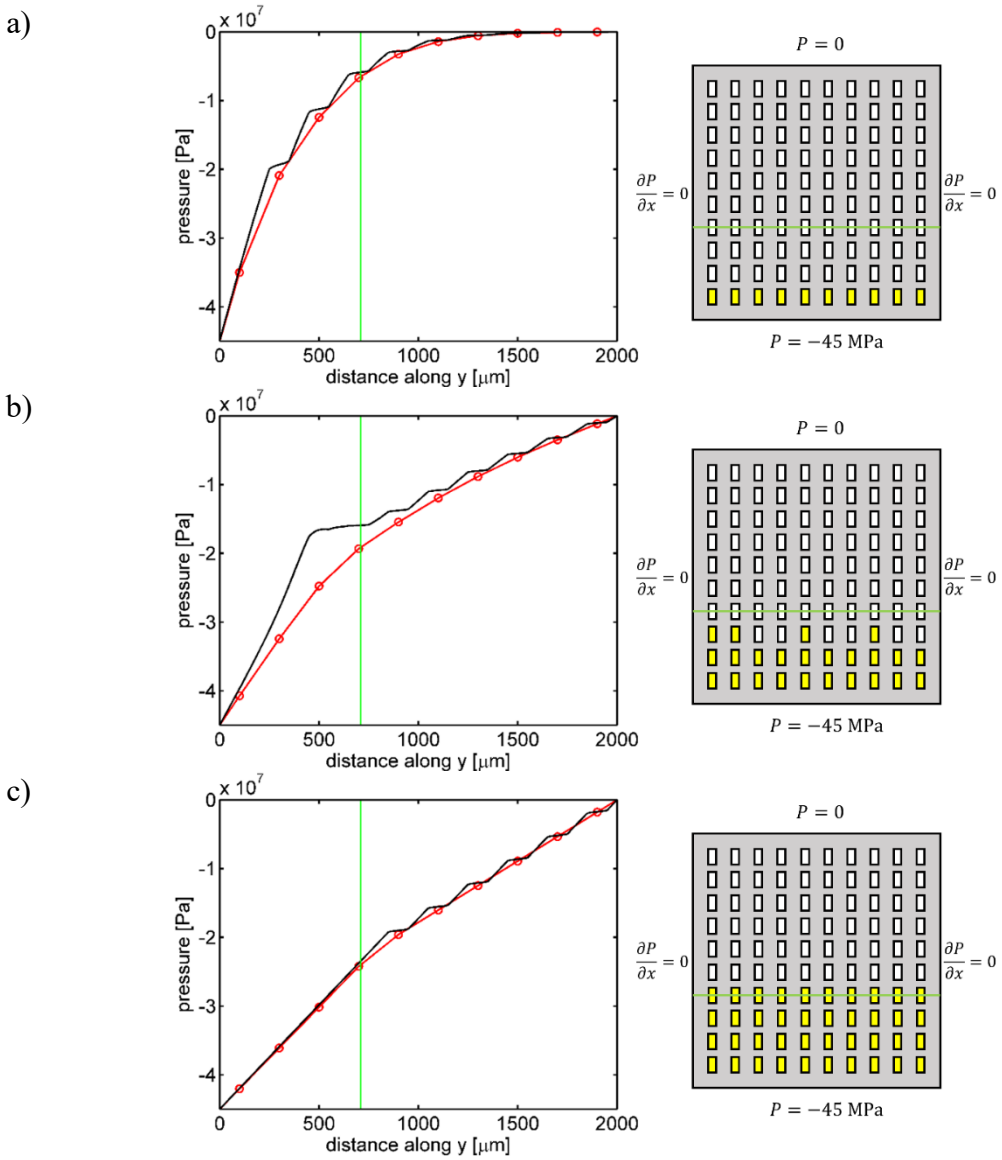


Figure 31. (Left) Transient pressure response during which cavitation can occur due to larger stress imposed at bottom boundary ($P_b = -45$ MPa $< P_{50}$). Pressure profile in the y-direction across the global domain at times (a) 1 s, (b) 5 s, and (c) 100 s, using the microscale model, averaged in the x-direction (black), and Maxwell's elliptic effective medium model (red). Since the boundary pressure at $y = 0$ is low enough to cause cavitation, the steady state location of $\mathcal{P}_{50}$ is marked in each plot (green). (Right) Schematic of geometry used in the microscale model and the corresponding boundary conditions applied to the global domain. Cavitated voids are marked yellow.

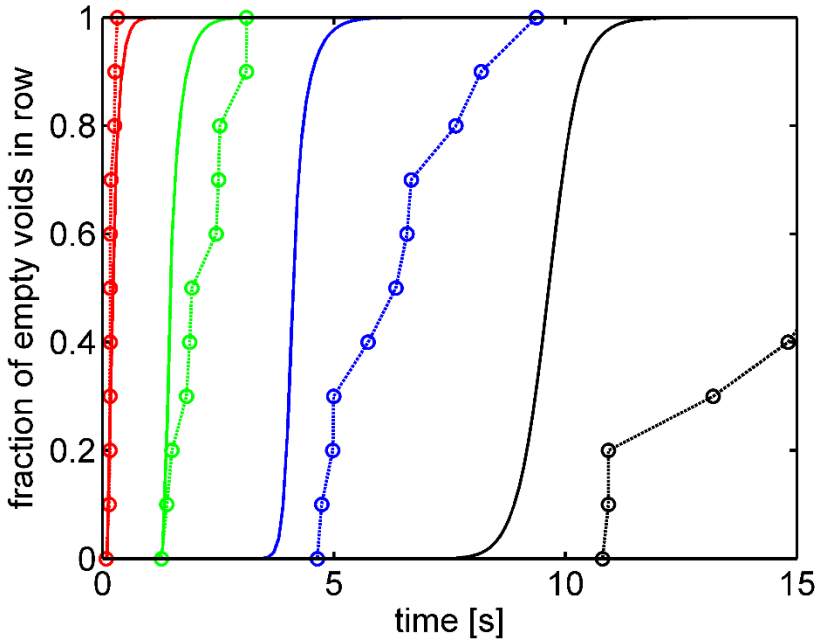


Figure 32. Fraction of empty voids in the first four rows in the microscale model shown in red, green, blue, and black (dashed lines with circular data points), respectively. These results correspond to the transient pressure simulation with cavitation, depicted in Figure 31. The solid lines correspond to the value of n_E in the effective medium model for the equivalent y-location.

4.4. Conclusion

Both the microscale and effective medium models presented here are able to solve for the pressure dynamics and mass transport of liquid through a nanoporous substrate containing microscale inclusions in the absence and presence of cavitation occurring within the domain. These predictions are important in contexts such as the drying rates of porous media as in fractured caprock or within biological tissues such as plant xylem. In cases without cavitation, these models agree closely with one another, and in such cases, the effective medium model is recommended because it is computationally more efficient and it provides a set of equations that are easier to

modify to accommodate diverse geometries, whereas the microscale model requires a mesh that discretizes and resolves the entire domain and may become too cumbersome to handle voids of irregular shape and distribution.

In cases where cavitation is expected, the microscale model, as derived here, may perform better since this model has been developed and compared to some experimental data. However, there remains a need for additional experimental data to help validate the results predicted here in the case of drying from a single edge of the domain and in cases where voids are elongated in the direction of drying. Furthermore, additional data could also be used to improve the effective medium model by modifying the mass transfer coefficient to fit experimental results of emptying times.

Finally, we note that these models assume that the larger, interior voids are only connected hydraulically with one another through the nanoporous substrate. In natural porous media, large fractures could exist that span the entire domain and in that case may dominate the transport of liquid in the region around such features. Such geometries could not be handled by the effective medium model because this model requires that any heterogeneous inclusions (voids in this case) have not percolated the network. However, such geometries could be handled by the microscale model provided that an explicit value of k_1 were used in the equations, instead of using the simplified equations derived by assuming $k_1/k_0 \gg 1$.

CHAPTER 5

CONCLUSIONS

The physics of multiphase flow in porous media are important in many natural and engineering contexts. In particular, as the United States considers imposing regulatory limitations to carbon dioxide emissions, geologic carbon sequestration has emerged as one possible compromise that would allow the continuation of fossil fuels to be the primary energy source while reducing anthropogenic atmospheric emissions by injecting the created CO2 deep underground. The injecting of CO2 into subsurface reservoirs is difficult to model because of the heterogeneity present throughout the domain and the lack of data available to characterize the properties required to predict the fluid dynamics, such as the saturation of other fluids present and the relative permeability of each fluid phase.

In this work, we presented our analysis of three important aspects of carbon sequestration. First, we analyzed an injection strategy using CO2 and brine that could be used to dramatically reduce the mobility of CO2 following a sequestration process; this outcome is important because post injection migration of CO2 (including lateral migration) increases the risk of CO2 escaping from the reservoir and returning to the surface. Second, we investigated the propensity for residually trapped CO2 to become remobilized and migrate was investigated. While those results indicated that even large pressure disturbances within the reservoir caused only a modest fraction of CO2 to become mobile, they also highlighted the crucial role that the model of relative permeability can play in affecting the predicted dynamics of the system. By including

percolation theory into the equations defining the relative permeability curves, not only was this new model in better agreement with experimental data, but in using it, the dynamics of remobilized CO2 were found to occur at least an order of magnitude faster than the traditional Brooks-Corey relative permeability curves (for typical Corey n values greater than 2.5). Finally, we presented an effective medium model and a microscale model to predict the dynamics of drying, motivated by the idea that dry CO2 would create a gradient of water potential between the CO2 phase and the caprock, causing the caprock to drain. Following experimental work in this theme, two different models were developed to characterize the pressure dynamics of heterogeneous porous media and to include the physics of cavitation.

Across these themes, there remains a broad need for well documented, experimental data on effective permeability, capacity, and relative permeability of micro and nanoporous media containing multiple fluid phases or heterogeneous inclusions or both. Preferably, this data would come from both the testing of natural porous media (despite the challenges involved) and from engineered porous media whose geometry is simple to characterize such that explicit computational models could be developed to match and compare results.

Nevertheless, the models and analysis techniques included here encompass a generalized set of tools and equations that may be used to predict the multiphase fluid dynamics of invasion, remobilization, and cavitation occurring in porous media. More importantly, these results provide crucial guidelines for the safe, long-term storage of carbon in the geologic subsurface. By understanding the fluid dynamics of CO2 injection and the interaction of CO2 with the surrounding bring and caprock, we may

guide policies related to encouraging brine flooding to decrease CO2 mobility and monitoring reservoir pore pressure to identify potential remobilization events or caprock drying by cavitation. These guidelines will be important in framing policy around long-term site stewardship to reduce the risk of CO2 migrating back toward the surface and negating our efforts to suppress rising CO2 concentrations in the atmosphere.

APPENDIX

Appendix 1

I. Origin of the J-Leverett Function and its connection to permeability

Origin of the Leverett J-Function

Early work involved static experiments of a porous medium using the fluids water and air. The porous medium was placed into a bath of water and the water was allowed to rise in the medium until equilibrium was reached. In a single column, the capillary pressure is related to the height of the liquid column and the Young-Laplace through (1)

$$P_c = P_{nw} - P_w = \rho_w g h = \frac{2\sigma \cos\theta}{R}$$

where σ is the surface tension between the wetting and nonwetting phases, θ is the contact angle, and R is the radius of the column.

Starting from this simplistic single-column relationship, Smith et. al. (1931) performed experiments with a porous medium comprised of uniform glass spheres. The spheres were randomly packed into a large bed. On a local scale, the spheres exhibited one of three different packing structures. The global configuration could then be presented as a probability of these three lattice structures. They allowed water to rise in the porous medium until equilibrium was reached. They concluded (through experiment) the following relationship

(2)

$$P_c = \rho_w g h = \frac{\sigma}{r} f(\phi) = \frac{\sigma}{r}\left[\frac{2}{\frac{0.9590}{(1-\phi)^{2/3}} - 1}\right]$$

where r is the radius of the spheres and ϕ is the porosity of the packed bed, which is some function of the probabilities of the three lattice configurations present. However, Smith did not related capillary pressure to saturation.

Leverett (1940) performed similar experiments to relate water height (capillary pressure) to water saturation in packed beds of sand. Leverett sites Smith's work but states that because real geologic samples contain complex internal geometries, no such relationship could be derived to describe the capillary pressure as a function of geometric properties or saturation, *a priori*. Unlike Smith, he used real sands as the porous matrix and water-oil as the two fluids. Measure h vs. S_w, he found that when he plotted $\Delta\rho g h \left(\sqrt{\frac{\kappa}{\phi}}\right)/\sigma$ vs. S_w for different sands, all data sets collapsed onto two curves: one for drainage and one for imbibition. This finding led to the Leverett J-function, written as

(3)

$$P_c(S_w) = \Delta\rho g h = \sigma \cos\theta \sqrt{\frac{\phi}{\kappa}} \mathcal{J}(S_w)$$

(Often $\cos\theta$ is omitted, as it becomes incorporated into the description of $\mathcal{J}(S_w)$). It has since been shown that Leverett functions did a good job describing the capillary pressure vs. saturation for a particular rock type, but one Leverett function could not be

used universally to describe multiple rock types. We can see that equation 3 is similar in form to equation 2 if we consider that $r \sim 1/\sqrt{\kappa}$.

Theory connecting permeability and Leverett J-function

From searching through literature, I have pieced together the following connecting-theory. Most of this comes from Noaman El-Khatib in an SPE conference proceeding (1995). El-Khatib's work involved making small improvements to the Leverett function in an attempt to collapse all Leverett J-functions onto a single function, which he calls the "modified J-function."

For a single tube, the flow rate is described by Hagen-Poiseuille as

(4)

$$Q = \frac{\pi R^4 \Delta P}{8 \mu L_e} = \frac{A_f R^2 \Delta P}{8 \mu L_e}$$

where $A_f = \pi R^2$ and L_e is the length of the flow path. From Darcy's law, the flow rate is

(5)

$$Q = \frac{A_{tot} \kappa \Delta P}{\mu L}$$

where L is the length of the porous medium. Using these, the permeability can be written as

(6)

$$\kappa = \frac{A_f}{A_{tot}} \frac{R^2}{8\left(\frac{L_e}{L}\right)}$$

Additionally, we know

(7)

$$V_{pores} = A_f L_e = A_{tot} L \phi$$

Using the relationship in equation 7, equation 6 can be written as

(8)

$$\kappa = \phi \left(\frac{R^2}{8\tau}\right)$$

where $\tau = (L_e/L)^2$.

If we now consider multiple, parallel cylinders with different pore radii and saturations, equation 8 can be written as

(9)

$$\kappa = \frac{\phi}{8\tau} \sum_i R_i^2 \Delta S_{wi}$$

The pore radii can now be written in terms of the capillary pressure, using Young-Laplace, given by

(10)

$$\kappa = \frac{\phi}{2\tau} \sum_i \left(\frac{\sigma \cos\theta}{P_c}\right)^2 \Delta S_{wi}$$

Now we assume that the porous media is comprised of an infinite distribution of pores, such that equation 10 becomes

(11)

$$\kappa = \frac{\phi}{2\tau}\int_0^1 (\sigma\cos\theta)^2 \frac{dS_w}{P_c^2}$$

Equation 11 can be solved for 2τ, which after some algebra, becomes

(12)

$$2\tau = \int_0^1 \left(\sigma\cos\theta\sqrt{\frac{\phi}{\kappa}}\right)^2 \frac{dS_w}{P_c^2} = \int_0^1 \frac{dS_w}{\mathcal{J}(S_w)^2}$$

Equation 12 now helps highlight the reason why one Leverett function fits reasonably well for a single rock type (same tortuosity) but not very well for multiply rock types.

El-Khatib went on to try and bring tau into the RHS of equation 12, in order to create a single, universal Leverett function, but was unable to do so agreeably.

Appendix 2

II. Deriving the volume ratio (CO2:brine) equations in Table 2

First Cycle – Trapping Ignoring Dissolution

Initial mass of CO2

$$m_{ci} = \phi\pi R_i^2 H \rho_c S_c^+$$

Final mass of CO2

$$m_{cf} = \phi\pi R_f^2 H \rho_c S_c^-$$

These masses must be equal, through COM, so use this to find the relationship with R

$$\phi\pi R_i^2 H \rho_c S_c^+ = \phi\pi R_f^2 H \rho_c S_c^-$$

$$R_i^2 \frac{S_c^+}{S_c^-} = R_f^2$$

The volume of brine is the final brine volume less the initial brine volume

$$V_b = R_f^2(1 - S_c^-) - R_i^2(1 - S_c^+)$$

Take ratio and substitute the R result from above, and simplify

$$\frac{V_b}{V_c} = \frac{R_f^2(1 - S_c^-) - R_i^2(1 - S_c^+)}{R_i^2 S_c^+}$$

$$\frac{V_b}{V_c} = \frac{S_c^+(1 - S_c^-) - S_c^-(1 - S_c^+)}{S_c^+ S_c^-}$$

$$\frac{V_b}{V_c} = \frac{S_c^+ - S_c^-}{S_c^+ S_c^-}$$

First Cycle – Trapping

Initial mass of CO2

$$m_{ci} = \phi\pi R_i^2 H\rho_c S_c^+ + \phi\pi R_i^2 H(1 - S_c^+)C_{aq}^+$$

Final mass of CO2

$$m_{cf} = \phi\pi R_f^2 H\rho_c S_c^- + \phi\pi R_f^2 H(1 - S_c^-)C_{aq}^+$$

These masses must be equal, through COM, so use this to find the relationship with R

$$\phi\pi R_i^2 H\rho_c S_c^+ + \phi\pi R_i^2 H(1 - S_c^+)C_{aq}^+ = \phi\pi R_f^2 H\rho_c S_c^- + \phi\pi R_f^2 H(1 - S_c^-)C_{aq}^+$$

$$R_i^2(S_c^+ + (1 - S_c^+)\psi^+) = R_f^2(S_c^- + (1 - S_c^-)\psi^+)$$

$$\frac{R_i^2(S_c^+ + (1 - S_c^+)\psi^+)}{(S_c^- + (1 - S_c^-)\psi^+)} = R_f^2$$

The volume of brine is the final brine volume less the initial brine volume

$$V_b = R_f^2(1 - S_c^-) - R_i^2(1 - S_c^+)$$

Take ratio and substitute the R result from above, and simplify

$$\frac{V_b}{V_c} = \frac{R_f^2(1 - S_c^-) - R_i^2(1 - S_c^+)}{R_i^2(S_c^+ + (1 - S_c^+)\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{\frac{(S_c^+ + (1 - S_c^+)\psi^+)}{(S_c^- + (1 - S_c^-)\psi^+)}(1 - S_c^-) - (1 - S_c^+)}{(S_c^+ + (1 - S_c^+)\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{(1 - S_c^-)(S_c^+ + (1 - S_c^+)\psi^+) - (1 - S_c^+)(S_c^- + (1 - S_c^-)\psi^+)}{(S_c^- + (1 - S_c^-)\psi^+)(S_c^+ + (1 - S_c^+)\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{S_c^+ - S_c^-}{(S_c^- + \psi^+ - S_c^-\psi^+)(S_c^+ + \psi^+ - S_c^+\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{S_c^+ - S_c^-}{(S_c^- + (1 - S_c^-)\psi^+)(S_c^+ + (1 - S_c^+)\psi^+)}$$

When psi is zero

$$\frac{V_b}{V_c} = \frac{S_c^+ - S_c^-}{S_c^+ S_c^-}$$

All Following Cycles – Trapping

The mass of CO2 injected is equal to the

$$m_{ci} = \phi\pi R_i^2 H \rho_c S_c^+$$

Final mass of CO2

$$m_{cf} = \phi\pi R_f^2 H \rho_c S_c^- + \phi\pi R_f^2 H (1 - S_c^-) C_{aq}^+$$

These masses must be equal, through COM, so use this to find the relationship with R

$$\phi\pi R_i^2 H \rho_c S_c^+ = \phi\pi R_f^2 H \rho_c S_c^- + \phi\pi R_f^2 H (1 - S_c^-) C_{aq}^+$$

$$R_i^2 S_c^+ = R_f^2 (S_c^- + (1 - S_c^-)\psi^+)$$

$$\frac{R_i^2 S_c^+}{(S_c^- + (1 - S_c^-)\psi^+)} = R_f^2$$

The volume of brine is the final brine volume less the initial brine volume

$$V_b = R_f^2 (1 - S_c^-) - R_i^2 (1 - S_c^+)$$

Take ratio and substitute the R result from above, and simplify

$$\frac{V_b}{V_c} = \frac{R_f^2 (1 - S_c^-) - R_i^2 (1 - S_c^+)}{R_i^2 S_c^+}$$

$$\frac{V_b}{V_c} = \frac{S_c^+ (1 - S_c^-) - (1 - S_c^+)(S_c^- + (1 - S_c^-)\psi^+)}{S_c^+ (S_c^- + (1 - S_c^-)\psi^+)}$$

When psi is zero

$$\frac{V_b}{V_c} = \frac{S_c^+ (1 - S_c^-) - S_c^- (1 - S_c^+)}{S_c^+ S_c^-}$$

First Cycle – Dissolution

Initial mass of CO2

$$m_{ci} = \phi\pi R_i^2 H\rho_c S_c^+ + \phi\pi R_i^2 H(1 - S_c^+)C_{aq}^+$$

Final mass of CO2

$$m_{cf} = \phi\pi R_f^2 H C_{aq}^+$$

These masses must be equal, through COM, so use this to find the relationship with R

$$\phi\pi R_i^2 H\rho_c S_c^+ + \phi\pi R_i^2 H(1 - S_c^+)C_{aq}^+ = \phi\pi R_f^2 H C_{aq}^+$$

$$R_i^2(S_c^+ + (1 - S_c^+)\psi^+) = R_f^2\psi^+$$

$$\frac{R_i^2(S_c^+ + (1 - S_c^+)\psi^+)}{\psi^+} = R_f^2$$

The volume of brine is the final brine volume less the initial brine volume

$$V_b = R_f^2 - R_i^2(1 - S_c^+)$$

Take ratio and substitute the R result from above, and simplify

$$\frac{V_b}{V_c} = \frac{R_f^2 - R_i^2(1 - S_c^+)}{R_i^2(S_c^+ + (1 - S_c^+)\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{(S_c^+ + (1 - S_c^+)\psi^+) - (1 - S_c^+)\psi^+}{\psi^+(S_c^+ + (1 - S_c^+)\psi^+)}$$

$$\frac{V_b}{V_c} = \frac{S_c^+}{\psi^+(S_c^+ + (1 - S_c^+)\psi^+)}$$

Appendix 3

III. Deriving the CO2 Saturation Wave Equation (Eq. 34)

Conservation of Mass – CO2

$$\frac{\partial}{\partial t}[\phi \rho_c S_c] + \boldsymbol{\nabla} \cdot (\rho_c \boldsymbol{u}_{dc}) = -hA_{cb}\phi S_b\left(C_{aq}^{+} - C_{aq}\right)$$

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\boldsymbol{\nabla} \cdot \boldsymbol{u}_{dc} = -\frac{hA_{cb}S_b\left(C_{aq}^{+} - C_{aq}\right)}{\rho_c} \tag{1}$$

Conservation of Mass – Brine

$$\frac{\partial}{\partial t}[\phi \rho_b S_b] + \boldsymbol{\nabla} \cdot (\rho_b \boldsymbol{u}_{db}) = 0$$

$$\frac{\partial S_b}{\partial t} + \frac{1}{\phi}\boldsymbol{\nabla} \cdot \boldsymbol{u}_{db} = 0 \tag{2}$$

Conservation of Mass – Aqueous CO2

$$\frac{\partial}{\partial t}[\phi S_b C] + \boldsymbol{\nabla} \cdot (C\boldsymbol{u}_{db}) = hA_{cb}\phi S_b\left(C_{aq}^{+} - C_{aq}\right)$$

$$\frac{\partial C}{\partial t} + \frac{\boldsymbol{u}_{dc}}{\phi S_b} \cdot \boldsymbol{\nabla} C = hA_{cb}\left(C_{aq}^{+} - C_{aq}\right) \tag{3}$$

To find the pressure gradient, we add equations (1) and (2) and recognize that

$\frac{\partial}{\partial t}[S_c + S_b] = 0$, which yields:

$$\boldsymbol{\nabla} \cdot (\boldsymbol{u}_{dc} + \boldsymbol{u}_{db}) = -D$$

where $D = (hA_{cb}\phi S_b \Delta C)/\rho_c$. In radial coordinates, this becomes

$$\frac{1}{r}\frac{\partial}{\partial r}[r(u_{dc} + u_{db})] = -D \tag{4}$$

Solve for the sum of Darcy velocities:

$$u_{dc} + u_{db} = \frac{1}{r}\int -rD\, dr \tag{5}$$

Although I don't know what this integral will be, I know that when the integral is evaluated, there will be an arbitrary constant that will need to be found by considering the condition at the well ($r = r_o$) where the sum of the Darcy velocities must be equal to $Q_{tot}/(2\pi H r_o)$. Here, I let $F(r) = \int -rD\ dr$ such that at $r = r_o$, equation (5) becomes:

$$u_{dc} + u_{db} = \frac{1}{r_o}(F(r_o) + k_1) = \frac{Q_{tot}}{2\pi H r_o} \tag{6}$$

Where k_1 is the arbitrary constant from integration, and can now be written as:

$$k_1 = \frac{Q_{tot}}{2\pi H} - F(r_o) \tag{7}$$

Equation (5) can now be written as

$$u_{dc} + u_{db} = \frac{1}{r}[F(r) + k_1]$$

$$u_{dc} + u_{db} = \frac{1}{r}\left[F(r) + \frac{Q_{tot}}{2\pi H} - F(r_o)\right]$$

$$u_{dc} + u_{db} = \frac{Q_{tot}}{2\pi H r} + \frac{1}{r}\int_{r_o}^{r} -r' D dr' \tag{8}$$

Equation (8) can be used to find the pressure gradient, and then the pressure gradient can be substituted back into each Darcy velocity. Doing so gives:

$$u_{dc} = \left[\frac{1}{1+\eta f}\right]\left[\frac{Q_{tot}}{2\pi H r} + \frac{1}{r}\int_{r_o}^{r} -r' D dr'\right] \tag{9}$$

$$u_{db} = \left[1 - \frac{1}{1+\eta f}\right]\left[\frac{Q_{tot}}{2\pi H r} + \frac{1}{r}\int_{r_o}^{r} -r' D dr'\right] \tag{10}$$

Going back now to equation (1)

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\nabla \cdot u_{dc} = -\frac{hA_{cb}S_b(C^* - C)}{\rho_c} \tag{11}$$

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\left[\frac{1}{r}\frac{\partial}{\partial r}(r u_{dc})\right] = -\frac{D}{\phi} \tag{12}$$

From here we must use the chain rule, Leibniz rule, and significant algebra to transform equation 12 into an equation where the method of characteristics can be applied. The end result is the following PDE.

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr} + \frac{1}{r}\int_{r_o}^{r} -r'Ddr'\right)\frac{\partial S_c}{\partial r} = \frac{D}{\phi}\left(\frac{1}{1+\eta f} - 1\right) \tag{13}$$

Equation 13 can now be transformed using the method of characteristics into the following set of coupled ordinary differential equations:

$$\frac{dS_c}{dt} = \frac{D}{\phi}\left(\frac{1}{1+\eta f} - 1\right) \tag{14}$$

$$\frac{dr}{dt} = v_{sc} = \frac{1}{\phi}\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr} + \frac{1}{r}\int_{r_o}^{r} -r'Ddr'\right) \tag{15}$$

Show work to go from equation 12 to equation 13

$$\frac{\partial S_c}{\partial t} + \frac{1}{\phi}\left[\frac{1}{r}\frac{\partial}{\partial r}(ru_{dc})\right] = -\frac{D}{\phi} \tag{12}$$

Multiply everything by phi and use Chain rule on term 2

$$\phi\frac{\partial S_c}{\partial t} + \frac{u}{r} + \frac{\partial u}{\partial r} = -D$$

Substitute the Darcy velocity (from equation 9) into term 3 (du/dr).

$$\phi\frac{\partial S_c}{\partial t} + \frac{u}{r} + \frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\left(\frac{Q_{tot}}{2\pi Hr} + \frac{1}{r}\int_{r_o}^{r} -r'Ddr'\right)\right] = -D$$

Distribute the (1/(1+eta*f)) term and separate the d/dr operation

$$\phi\frac{\partial S_c}{\partial t} + \frac{u}{r} + \frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\frac{Q_{tot}}{2\pi Hr}\right] + \frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\int_{r_o}^{r} -r'Ddr'\right] = -D$$

Use chain rule on term 3, use chain rule on term 4

$$\phi\frac{\partial S_c}{\partial t}+\frac{u}{r}+\left(\frac{1}{1+\eta f}\right)\left(\frac{-Q_{tot}}{2\pi Hr^2}\right)+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]$$

$$+\left(\int_{r_o}^{r}-r'Ddr'\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\right]=-D$$

Terms 2 and 3 can be moved to the RHS of the equation, and the chain rule can be used on term 6

$$\phi\frac{\partial S_c}{\partial t}+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]$$

$$+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\left(\frac{1}{1+\eta f}\right)\left(-\frac{1}{r^2}\right)+\left(\frac{1}{r}\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\right]\right)$$

$$=-D-\frac{u}{r}+\left(\frac{1}{1+\eta f}\right)\left(\frac{Q_{tot}}{2\pi Hr^2}\right)$$

Distribute the integral in term 4

$$\phi\frac{\partial S_c}{\partial t}+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{1+\eta f}\right)\left(-\frac{1}{r^2}\right)$$

$$+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{r}\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\right]=-D-\frac{u}{r}+\left(\frac{1}{1+\eta f}\right)\left(\frac{Q_{tot}}{2\pi Hr^2}\right)$$

Term 4 can be moved to the RHS of the equation

$$\phi\frac{\partial S_c}{\partial t}+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{r}\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\right]$$

$$=-D-\frac{u}{r}+\left(\frac{1}{1+\eta f}\right)\left(\frac{Q_{tot}}{2\pi Hr^2}\right)+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{1+\eta f}\right)\left(\frac{1}{r^2}\right)$$

On the RHS of the equation, we can factor out a (1/(1+eta*f)

$$\phi\frac{\partial S_c}{\partial t}+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{r}\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\right]$$

$$=-D-\frac{u}{r}+\left(\frac{1}{1+\eta f}\right)\left(\frac{Q_{tot}}{2\pi Hr^2}+\frac{1}{r^2}\int_{r_o}^{r}-r'Ddr'\right)$$

We now recognize on the RHS that term 2 (u/r) will cancel out with term 3

$$\phi\frac{\partial S_c}{\partial t}+\frac{Q_{tot}}{2\pi Hr}\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]+\left(\int_{r_o}^{r}-r'Ddr'\right)\left(\frac{1}{r}\right)\frac{\partial}{\partial r}\left[\left(\frac{1}{1+\eta f}\right)\right]=-D$$

Term 2 and term 4 can be combined, since they share the common factor of d/dr(1/(1+eta*f))

$$\phi\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr}+\frac{1}{r}\int_{r_o}^{r}-r'Ddr'\right)+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}\frac{\partial}{\partial r}\left[\int_{r_o}^{r}-r'Ddr'\right]=-D$$

We now apply Leibniz rule on term 3 to bring the derivative into the integral

$$\phi\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr}+\frac{1}{r}\int_{r_o}^{r}-r'Ddr'\right)+\left(\frac{1}{1+\eta f}\right)\frac{1}{r}(-rD)=-D$$

Move term 3 to the RHS of the equation

$$\phi\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial r}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr}+\frac{1}{r}\int_{r_o}^{r}-r'Ddr'\right)=-D+\left(\frac{1}{1+\eta f}\right)D$$

Divide through by phi, and recast term 2 for the method of characteristics

$$\frac{\partial S_c}{\partial t}+\frac{1}{\phi}\frac{\partial}{\partial S_c}\left[\frac{1}{1+\eta f}\right]\left(\frac{Q_{tot}}{2\pi Hr}+\frac{1}{r}\int_{r_o}^{r}-r'Ddr'\right)\frac{\partial S_c}{\partial r}=\frac{D}{\phi}\left(\frac{1}{1+\eta f}-1\right)$$

The above line is now equation (13)

Appendix 4

IV. Deriving the nondimensional CO2 mass conservation in the z-direction (Eq. 72)

The general conservation of mass for a fluid in the absence of dissolution and chemical reaction is given by

(1)
$$\frac{\partial}{\partial t}[\phi \rho_i S_i] + \frac{\partial}{\partial z}[\rho_i u_{di}] = 0$$

Given a system of only CO2 and brine, assuming each phase is incompressible, equation (1) can be written for each phase as

(2)
$$\phi \frac{\partial S_c}{\partial t} + \frac{\partial}{\partial z}[u_{dc}] = 0$$

(3)
$$\phi \frac{\partial S_b}{\partial t} + \frac{\partial}{\partial z}[u_{db}] = 0$$

Equations (2) and (3) are added together to yield

(4)
$$\phi \frac{\partial}{\partial t}[S_c + S_b] + \frac{\partial}{\partial z}[u_{dc} + u_{db}] = 0$$

The left hand term is equal to zero because we assume that CO2 and brine are the only two fluids in the pores. Using the definition of Darcy's law, equation (4) can be written as

(5)
$$\frac{\partial}{\partial z}\left[\frac{\kappa k_c}{\mu_c}\left(-\frac{\partial P_c}{\partial z}\right) - \frac{\kappa k_c}{\mu_c}\rho_c g + \frac{\kappa k_b}{\mu_b}\left(-\frac{\partial P_b}{\partial z}\right) - \frac{\kappa k_b}{\mu_b}\rho_b g\right] = 0$$

κ can be divided out and the brine pressure gradient can be substituted with $-\frac{\partial P_b}{\partial z} = -\frac{\partial P_c}{\partial z} + \frac{\partial P_{cap}}{\partial z}$. From this, equation (5) can be written as

(6)

$$\frac{\partial}{\partial z}\left[\frac{k_c}{\mu_c}\left(-\frac{\partial P_c}{\partial z}\right)-\frac{k_c}{\mu_c}\rho_c g+\frac{k_b}{\mu_b}\left(-\frac{\partial P_c}{\partial z}+\frac{\partial P_{cap}}{\partial z}\right)-\frac{k_b}{\mu_b}\rho_b g\right]=0$$

Equation (6) is multiplied by μ_c to give

(7)

$$\frac{\partial}{\partial z}\left[(k_c+\eta k_b)\left(-\frac{\partial P_c}{\partial z}\right)-k_c\rho_c g+\eta k_b\frac{\partial P_{cap}}{\partial z}-\eta k_b\rho_b g\right]=0$$

Equation (7) is integrated with respect to z. However, because we assume no net flux across the reservoir boundary, this becomes

(8)

$$(k_c+\eta k_b)\left(-\frac{\partial P_c}{\partial z}\right)-k_c\rho_c g+\eta k_b\frac{\partial P_{cap}}{\partial z}-\eta k_b\rho_b g=0$$

From equation (8), the CO2 pressure gradient can be written as

(9)

$$\left(-\frac{\partial P_c}{\partial z}\right)=\frac{1}{k_c+\eta k_b}\left(k_c\rho_c g+\eta k_b\rho_b g-\eta k_b\frac{\partial P_{cap}}{\partial z}\right)$$

Equation (9) is substituted into equation (2) to give

(10)

$$\phi\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial z}\left[\frac{\kappa k_c}{\mu_c}\left(\frac{1}{k_c+\eta k_b}\left(k_c\rho_c g+\eta k_b\rho_b g-\eta k_b\frac{\partial P_{cap}}{\partial z}\right)\right)-\frac{\kappa k_c}{\mu_c}\rho_c g\right]=0$$

At this point, significant algebra is required to write equation (10) as the final result required,

(11)

$$\frac{\partial S_c}{\partial \tilde{t}}+\frac{\partial}{\partial \tilde{z}}\left[k_c\frac{\eta f}{1+\eta f}\left(1-\frac{1}{Bo}\frac{d\mathcal{J}(S_c)}{dS_c}\frac{\partial S_c}{\partial \tilde{z}}\right)\right]=0$$

These steps are shown below.

Equation (10)

$$\phi\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial z}\left[\frac{\kappa k_c}{\mu_c}\left(\frac{1}{k_c+\eta k_b}\left(k_c\rho_c g+\eta k_b\rho_b g-\eta k_b\frac{\partial P_{cap}}{\partial z}\right)\right)-\frac{\kappa k_c}{\mu_c}\rho_c g\right]=0$$

(12)

$$\frac{\phi\mu_c}{\kappa}\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial z}\left[\left(\frac{k_c}{k_c+\eta k_b}\left(k_c\rho_c g+\eta k_b\rho_b g-\eta k_b\frac{\partial P_{cap}}{\partial z}\right)\right)-k_c\rho_c g\right]=0$$

(13)

$$\frac{\phi\mu_c}{\kappa}\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial z}\left[\frac{k_c}{k_c+\eta k_b}k_c\rho_c g+\frac{k_c}{k_c+\eta k_b}\eta k_b\rho_b g-\frac{k_c}{k_c+\eta k_b}\eta k_b\frac{\partial P_{cap}}{\partial z}-k_c\rho_c g\right]=0$$

(14)

$$\frac{\phi\mu_c}{\kappa}\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial z}\left[\left(\frac{k_c}{k_c+\eta k_b}-1\right)k_c\rho_c g+\frac{k_c}{k_c+\eta k_b}\eta k_b\rho_b g-\frac{k_c}{k_c+\eta k_b}\eta k_b\frac{\partial P_{cap}}{\partial z}\right]=0$$

(15)

$$\frac{\phi h^2\mu_c}{\kappa}\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial\tilde{z}}\left[\left(\frac{k_c}{k_c+\eta k_b}\right)k_c\rho_c gh-k_c\rho_c gh+\frac{k_c}{k_c+\eta k_b}\eta k_b\rho_b gh-\frac{k_c}{k_c+\eta k_b}\eta k_b(\Delta\rho gh)\frac{\partial\tilde{P}_{cap}}{\partial\tilde{z}}\right]=0$$

(16)

$$\frac{\phi h^2\mu_c}{\kappa\Delta\rho gh}\frac{\partial S_c}{\partial t}+\frac{\partial}{\partial\tilde{z}}\left[\left(\frac{k_c}{k_c+\eta k_b}\right)\frac{k_c\rho_c}{\Delta\rho}-\frac{k_c\rho_c}{\Delta\rho}+\frac{k_c}{k_c+\eta k_b}\eta\frac{k_b\rho_b}{\Delta\rho}-\frac{k_c}{k_c+\eta k_b}\eta k_b\frac{\partial\tilde{P}_{cap}}{\partial\tilde{z}}\right]=0$$

Here, the nondimensional time is defined as

$$\tilde{t}=\frac{\kappa\Delta\rho gh}{\phi\mu_c h^2}t$$

Equation (16) is now written as

(17)

$$\frac{\partial S_c}{\partial\tilde{t}}+\frac{\partial}{\partial\tilde{z}}\left[\frac{\rho_c k_c}{\Delta\rho}\left(\frac{1}{1+\eta f}-\left(\frac{1+\eta f}{1+\eta f}\right)+\frac{\eta f}{1+\eta f}\frac{\rho_b}{\rho_c}-\frac{\eta f}{1+\eta f}\frac{\Delta\rho}{\rho_c}\frac{\partial\tilde{P}_{cap}}{\partial\tilde{z}}\right)\right]=0$$

(18)

$$\frac{\partial S_c}{\partial\tilde{t}}+\frac{\partial}{\partial\tilde{z}}\left[\frac{\rho_c k_c}{\Delta\rho}\left(\frac{-\eta f}{1+\eta f}+\frac{\eta f}{1+\eta f}\frac{\rho_b}{\rho_c}-\frac{\eta f}{1+\eta f}\frac{\Delta\rho}{\rho_c}\frac{\partial\tilde{P}_{cap}}{\partial\tilde{z}}\right)\right]=0$$

(19)

$$\frac{\partial S_c}{\partial\tilde{t}}+\frac{\partial}{\partial\tilde{z}}\left[\frac{\rho_c k_c}{\Delta\rho}\left(\frac{\eta f}{1+\eta f}\left(\frac{\rho_b}{\rho_c}-1\right)-\frac{\eta f}{1+\eta f}\frac{\Delta\rho}{\rho_c}\frac{\partial\tilde{P}_{cap}}{\partial\tilde{z}}\right)\right]=0$$

(20)

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[\frac{\rho_c k_c}{\Delta\rho}\left(\frac{\eta f}{1+\eta f}\left(\frac{\rho_b}{\rho_c} - \frac{\rho_c}{\rho_c}\right) - \frac{\eta f}{1+\eta f}\frac{\Delta\rho}{\rho_c}\frac{\partial \tilde{P}_{cap}}{\partial \tilde{z}}\right)\right] = 0$$

(21)

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[\frac{\rho_c k_c}{\Delta\rho}\left(\frac{\eta f}{1+\eta f}\left(\frac{\Delta\rho}{\rho_c}\right) - \frac{\eta f}{1+\eta f}\frac{\Delta\rho}{\rho_c}\frac{\partial \tilde{P}_{cap}}{\partial \tilde{z}}\right)\right] = 0$$

(22)

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[k_c\frac{\eta f}{1+\eta f}\left(1 - \frac{\partial \tilde{P}_{cap}}{\partial \tilde{z}}\right)\right] = 0$$

(23)

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[k_c\frac{\eta f}{1+\eta f}\left(1 - \frac{\partial \tilde{P}_{cap}}{\partial S_c}\frac{\partial S_c}{\partial \tilde{z}}\right)\right] = 0$$

Using the Leverett-J function, equation (23) can be written as the final result
(24)

$$\frac{\partial S_c}{\partial \tilde{t}} + \frac{\partial}{\partial \tilde{z}}\left[k_c\frac{\eta f}{1+\eta f}\left(1 - \frac{1}{Bo}\frac{d\mathcal{J}(S_c)}{dS_c}\frac{\partial S_c}{\partial \tilde{z}}\right)\right] = 0$$

This is equation (11) listed above.

Appendix 5

V. Matlab codes to solve equations 34 and 35

```
% Solving for CO2 Injection with dissolution in cylindrical
coordinates
% Last Edited 12 February 2015 by Erik J. Huber

clear all; close all; clc
filename=('C1');

%% INPUTS
rhoc=850;                          %CO2 density
rhob=1010;                         %brine density
eta=0.1;                           %viscosity ratio
phi=0.15;                          %porosity
H=10;                              %reservoir height

Sn=0.25;                           %Scmin
Sx=0.8;                            %Scmax
kcx=1.0;
kbx=1.0;
Coreyn=1;                          %CO2 exponent
Coreym=1;                          %brine exponent

%Dissolution Inputs
Caqsat=60;                         %maximum saturation of CO2 in brine
[kg CO2 m^-3 brine]
h=1e-4;                            %mass transfer coefficient

% Pumping Inputs
%Pump-Switch-Matrix (PSM) [1st switch time, 2nd switch time, ....
switch OFF, DUMMY TIME]
%All pumping starts with co2 at time=0!!
%Enter PSM times in DAYS
PSM=[90,2362,1e5];
PSM_index=1;
PSM(1,:)=3600*24*PSM(1,:);          %convert days to seconds
current_fluid=1;                    %CO2=1, brine=-1
mdot=3;                             %Mass flowrate of CO2 in
reservoir [kg s^-1]
Qtot=mdot/rhoc;                     %Flowrate entering reservoir [m^3
s^-1]

% Code Inputs
r0=6;                               %well radius - start of
simulation
default_dt=150;                     %this is the default dt [s]
min_dx=0.01;                       %neighbors are too close
max_dx=0.1;                         %neighbors are too far apart
```

```
min_df=1e-5;                          %neighbors have the same approx
the same value
max_df_Sc=1e-3;
max_df_Aq=0.5;
Sc_near_zero=1e-6;                    %as CO2 dissolves, any packet
with Sc<this is considered "0"
Caq_near_sat=1e-6;                    %as CO2 dissolves, any packet
with Caq>Caqsat-this is considered "Caqsat"

%outputs and saving
show_outputs=1;

show_ScCaq_plot=0;                    %show plot while running 1=YES
0=NO
show_ScCaq_plot_interval=01*3600;     %show plot every ______ [s]

show_mass_error=1;                    %show mass error 1=YES 0=NO
show_mass_error_plot=0;               %show mass error 1=YES 0=NO
store_mass_error=1;                   %Store an array to plot the mass
error 1=YES 0=NO
mass_error_interval=01*3600;          %calculate the mass error every
_____ [s]
PE=zeros(1,5);

movie_on=0;                              %store movie data 1=YES 0=NO
movie_interval=1*24*3600;                %capture movie data every
_____ [s]
workspace_save_interval=1*24*3600;       %save the workspace every
_____ [s]

fname=sprintf('%s_%i',filename,1);

%% Change strict pump switch times into slow gradients
%T = [time , saturation]
dt=60;
Scfront=0;
Scback=Sx;
di=100;
for i=0:di
    T(i+1,1)=i*dt;
    T(i+1,2)=Scfront-((i/di)*(Scfront-Scback));
end
for j=1:size(PSM,2)-1
    t=PSM(1,j);
    current_fluid=current_fluid*-1;
    if current_fluid==1; Sc=Sx; else Sc=Sn; end
    Scfront=T(end,2);
    Scback=Sc;
    for i=0:di
        T(end+1,1)=t+i*dt;
        T(end,2)=Scfront-((i/di)*(Scfront-Scback));
    end
end
Tindex=1;
```

```
%% Matrix Definitions

% FREE PHASE CO2 (S) and Aqueous Phase CO2 (A)
% 1 - r-location
% 2 - Phase ID (0=Free-phase, 1=Aqueous Phase)
% 3 - Particle ID (0=normal, 1=shock)
% 4 - Particle Velocity (vsc/vaq OR wsc/waq)
% 5 - Sc
% 6 - Caq
% 7 - Darcy Velocity (udc OR udb)
% 8 - Integrals
% 9 - Sc behind shock
%10 - Sc in front of shock
%11 - ud behind (udc for wsc, udb for waq)
%12 - ud front (udc for wsc, udb for waq)
%13 - Caq behind (for waq)
%14 - Caq front (for waq)
%15 - logic kill column
S=zeros(1,15);
A=zeros(1,15);

%% Problem Set Up and Initial Conditions
mass_co2_inj=0;
mass_co2_sc=0;
mass_co2_aq=0;
t=0;
dt=default_dt;

% FREE PHASE - Sc nodes
Sc=0;
k=1;
S(k,1)=r0;
S(k,5)=Sc;
S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));

% k=1;
% go=1;
% while go==1
%     S(k,1)=r0+(k-1)*min_dx;
%     S(k,5)=Sx-(k-1)*max_df;
%
S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));
%     if S(k,5)<=0
%         S(k,5)=0;
%         go=0;
%     end
%     k=k+1;
% end
% S=sortrows(S,-1);

% % Aqueous Phase - Aq node
```

```
k=1;
A(k,1)=r0;
A(k,2)=1;
A(k,5)=0;
A(k,7)=udb(Sc,Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,A(k,1),A(k,8));
A(k,4)=vaq(A(k,7),A(k,5),phi);

if movie_on==1;
    Movie_Frames{1,1}=t;
    Movie_Frames{2,1}=S;
    Movie_Frames{3,1}=A;
    Movie_Frames{4,1}=mass_co2_inj;
end

frame_m=1;                  %Movie Frame Counter
frame_s=1;                  %Workspace Save Counter
frame_p=0;                  %Plot figure counter
frame_me=1;                 %Mass Error counter
percent_error=0;
t_days=0;

%% Run the main code
steps=1;

while t<=PSM(1,end-1) %&& steps<1950
    steps=steps+1;

    h=0;
    if t>3600
        h=1e-4;
    end

    %% STEP 1-Find the smallest dt
    hit_dt=default_dt;
    min_dt=default_dt;
    %%
    %Check time to add next injection packet
    hit_dt=T(Tindex+1,1)-t;
    if hit_dt<=min_dt;
        min_dt=hit_dt;
    end
    %%
    %Check Sc falling to zero
    dissolve_dt=default_dt*ones(size(S,1),1);
    for i=1:size(dissolve_dt,1)
        if S(i,3)==0 && S(i,5)>0 && S(i,6)<Caqsat && h~=0
            Sc=S(i,5);
            if Sc<=Sn
                kc=0;
                kb=1;
            elseif Sc>=Sx
                kc=1;
                kb=0;
            else
```

```
            kc=(kcx)*(((Sc-Sn)/(Sx-Sn))^Coreyn);
            kb=(kbx)*(((Sx-Sc)/(Sx-Sn))^Coreym);
        end
        U=(1/(1+(eta*(kb/kc))))-1;
        if U~=0
            Y=h*(1-S(i,5))*(Caqsat-S(i,6))/rhoc;
            if U*Y<0
                dissolve_dt(i,1)=(-1*S(i,5))/(Y*U);
            end
        end
    end
end
hit_dt=min(dissolve_dt(:,1));
if hit_dt<=min_dt;
    min_dt=hit_dt;
end
%%
%Check Aq going to Caqsat
if h~=0
    hit_dt=1/h;
else
    hit_dt=default_dt;
end
if hit_dt<=min_dt;
    min_dt=hit_dt;
end
%%
%Check collisions
hit_flag=0;
hit=zeros(1,2);
%Free-Phase into Free-Phase
for i=1:size(S,1)-1
    j=i+1;
    if S(j,4)>S(i,4)
        hit_dt=(S(i,1)-S(j,1))/(S(j,4)-S(i,4));
        if hit_dt<=min_dt;
            min_dt=hit_dt;
            hit(1,2)=i;
            hit(1,1)=j;
            hit_flag=1;
        end
    end
end
%%
%Aq-phase into Aq-phase
for i=1:size(A,1)-1
    j=i+1;
    if A(j,4)>A(i,4)
        hit_dt=(A(i,1)-A(j,1))/(A(j,4)-A(i,4));
        if hit_dt<=min_dt;
            min_dt=hit_dt;
            hit(1,2)=i;
            hit(1,1)=j;
            hit_flag=2;
        end
```

```
        end
    end
    %%

    %Now check for shocks in the Sc-phase to either pass Aq-packets
or be
    %passed by Aq-packets. This should also limit the time step.
    for i=1:size(S,1)
        search_flag=0;
        if S(i,3)==1    %found a shock, now find the nearest Caq
packet
            for j=1:size(A,1)
                if A(j,1)<S(i,1)
                    break
                end
            end
            if j>1
                j=j-1;
                if A(j,3)==0
                    search_flag=1;
                end
            end
            %At this point, "i" is the shock row in S
            %and "j" is the first Aq row ahead of this shock (could
be
            %row 1)
            if search_flag==1
                if S(i,4)>A(j,4) && A(j,1)>S(i,1)
                    hit_dt=(A(j,1)-S(i,1))/(S(i,4)-A(j,4));
                    if hit_dt<min_dt
                        min_dt=hit_dt;
                        Scrow=i;
                        Aqrow=j;
                        hit_flag=3;
                    end
                end
            end
        end
    end
    %Check Aq packets passing Sc shocks. These Caq nodes will need
their
    %concentrations adjusted to compensate for the change in Sc.
    for i=1:size(S,1)
        search_flag=0;
        if S(i,3)==1    %found a shock, now find the nearest Caq
packet
            for j=1:size(A,1)
                if A(j,1)<S(i,1)
                    break
                end
            end
            %Here, any Aq packet is ok, including node 1
            if A(j,3)==0
                search_flag=1;
            end
```

```
            %At this point, "i" is the shock row in S
            %and "j" is the first Aq row behind this shock (could be
            %row 1)
            if search_flag==1
                if A(j,4)>S(i,4) && S(i,1)>A(j,1)
                    hit_dt=(S(i,1)-A(j,1))/(A(j,4)-S(i,4));
                    if hit_dt<min_dt
                        min_dt=hit_dt;
                        Scrow=i;
                        Aqrow=j;
                        hit_flag=4;
                    end
                end
            end
        end
    end

    %Advance time by the minimum dt
    dt=min_dt;
    t=t+dt;

    %Record the mass addition in this time step
    Sc=T(Tindex,2);

mass_co2_inj=mass_co2_inj+(rhoc*2*pi()*r0*H*dt*udc(Sc,Sn,Sx,kcx,kbx,C
oreyn,Coreym,eta,Qtot,H,r0,0));


    %% STEP 2- dSc/dt and dCaq/dt with and without correction

    %dSc/dt
    for i=1:size(S,1)
        if S(i,3)==0 && S(i,5)>0 && S(i,6)<Caqsat
            Sc=S(i,5);
            if Sc<=Sn
                kc=0;
                kb=1;
            elseif Sc>=Sx
                kc=1;
                kb=0;
            else
                kc=(kcx)*(((Sc-Sn)/(Sx-Sn))^Coreyn);
                kb=(kbx)*(((Sx-Sc)/(Sx-Sn))^Coreym);
            end
            U=(1/(1+(eta*(kb/kc))))-1;
            Y=h*(Caqsat-S(i,6))*(1-S(i,5))/rhoc;
            S(i,5)=Sc+(dt*Y*U);
        end
%         if S(i,5)<Sc_near_zero
%             S(i,5)=0;
%         end
    end
    %dCaq/dt
    for i=1:size(A,1)
```

```
        if A(i,3)==0 && A(i,5)>0 && A(i,6)<Caqsat
            A(i,6)=A(i,6)+(dt*h*(Caqsat-A(i,6)));
            if Caqsat - A(i,6) < Caq_near_sat
                A(i,6)=Caqsat;
            end
            if A(i,6)<0
                A(i,6)=0;
            end
        end
    end


    %% STEP 3-Update the position of all nodes

    A(:,1)=A(:,1)+(A(:,4)*dt);
    S(:,1)=S(:,1)+(S(:,4)*dt);


    %% STEP 4-Remove Particles if Collision Occurred

    if hit_flag~=0
        %Sc hits Sc
        if hit_flag==1
            %Sc PHASE
            Q=zeros(size(S,1)-1,size(S,2));
            B_row=hit(1,1); F_row=hit(1,2); k=1;
            for j=1:size(S,1); if j~=F_row && j~=B_row;
Q(k,:)=S(j,:); k=k+1; end; end; k=size(Q,1);

            if S(F_row,2)==0 && S(B_row,2)==0
                if S(F_row,3)==0 && S(B_row,3)==0
                    Q(k,1)=S(F_row,1);
                    Q(k,2)=0;
                    Q(k,3)=1;
                    Q(k,9)=S(B_row,5);
                    Q(k,11)=S(B_row,7);
                    Q(k,13)=S(B_row,6);
                    Q(k,10)=S(F_row,5);
                    Q(k,12)=S(F_row,7);
                    Q(k,14)=S(F_row,6);
                    Q(k,4)=wsc(Q(k,9),Q(k,10),Q(k,11),Q(k,12),phi);
                elseif S(F_row,3)==1 && S(B_row,3)==0
                    Q(k,1:14)=S(F_row,1:14);
                elseif S(F_row,3)==0 && S(B_row,3)==1
                    Q(k,1:14)=S(B_row,1:14);
                else
                    Q(k,1)=S(F_row,1);
                    Q(k,2)=0;
                    Q(k,3)=1;
                    Q(k,9)=S(B_row,9);
                    Q(k,11)=S(B_row,11);
                    Q(k,13)=S(B_row,13);
                    Q(k,10)=S(F_row,10);
                    Q(k,12)=S(F_row,12);
```

```
                    Q(k,14)=S(F_row,14);
                    Q(k,4)=wsc(Q(k,9),Q(k,10),Q(k,11),Q(k,12),phi);
                end
            end
            S=sortrows(Q,-1);
        end

        %Aq hits Aq
        if hit_flag==2
            %Aq PHASE
            Q=zeros(size(A,1)-1,size(A,2));
            B_row=hit(1,1); F_row=hit(1,2); k=1;
            for j=1:size(A,1); if j~=F_row && j~=B_row;
Q(k,:)=A(j,:); k=k+1; end; end; k=size(Q,1);

            if A(F_row,3)==0 && A(B_row,3)==0
                if A(F_row,6)==0 && A(B_row,6)==0
                    Q(k,1:14)=A(F_row,1:14);
                else
                    Q(k,1)=A(F_row,1);
                    Q(k,2)=1;
                    Q(k,3)=1;
                    Q(k,9)=A(B_row,5);
                    Q(k,11)=A(B_row,7);
                    Q(k,13)=A(B_row,6);
                    Q(k,10)=A(F_row,5);
                    Q(k,12)=A(F_row,7);
                    Q(k,14)=A(F_row,6);

Q(k,4)=waq(Q(k,9),Q(k,10),Q(k,11),Q(k,12),Q(k,13),Q(k,14),phi);
                end
            elseif A(F_row,3)==1 && A(B_row,3)==0
                Q(k,1:14)=A(F_row,1:14);
            elseif A(F_row,3)==0 && A(B_row,3)==1
                Q(k,1:14)=A(B_row,1:14);
            else
                Q(k,1)=A(F_row,1);
                Q(k,2)=1;
                Q(k,3)=1;
                Q(k,9)=A(B_row,9);
                Q(k,11)=A(B_row,11);
                Q(k,13)=A(B_row,13);
                Q(k,10)=A(F_row,10);
                Q(k,12)=A(F_row,12);
                Q(k,14)=A(F_row,14);

Q(k,4)=waq(Q(k,9),Q(k,10),Q(k,11),Q(k,12),Q(k,13),Q(k,14),phi);
            end
            A=sortrows(Q,-1);
        end

        %Sc shock passes Aq
        if hit_flag==3
            v1=A(Aqrow,4);
            C1=A(Aqrow,6);
```

```
            S1=A(Aqrow,5);
            S2=S(Scrow,9);

udb2=udb(S2,Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,S(Scrow,1),A(Aqrow
,8));
            v2=vaq(udb2,S2,phi);
            w=S(Scrow,4);

            A(Aqrow,6)=(C1*(1-S1)*(v1-w))/((1-S2)*(v2-w));
            if A(Aqrow,6)>Caqsat
                A(Aqrow,6)=Caqsat;
            end
        end

        %Aq passes Sc shock
        if hit_flag==4
            C1=A(Aqrow,6);
            S1=A(Aqrow,5);
            S2=S(Scrow,10);
            v1=A(Aqrow,4);

udb2=udb(S2,Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,S(Scrow,1),A(Aqrow
,8));
            v2=vaq(udb2,S2,phi);
            w=S(Scrow,4);

            A(Aqrow,6)=(C1*(1-S1)*(v1-w))/((1-S2)*(v2-w));
        end
    end


    %% STEP 5- Add packets to the location r=r0 if needed

    Sc=T(Tindex,2);
    if S(end,1)>r0+min_dx
        k=size(S,1);
        S(k+1,1:5)=[r0,0,0,0,Sc];
    end

    if A(end,1)>r0+min_dx
        k=size(A,1);
        A(k+1,1:6)=[r0,1,0,0,Sc,0];
    end



    %% STEP 6-Add packets into the simulation

    %Sc series
    k=size(S,1);
    for i=1:size(S,1)-1
        j=i+1;
        %Just plain old too far apart
```

```
    if S(i,3)==0 && S(j,3)==0
        if S(i,1)-S(j,1)>max_dx
            k=k+1;
            S(k,1)=(S(i,1)+S(j,1))/2;
            S(k,2)=0;
            S(k,5)=(S(i,5)+S(j,5))/2;
        end
    elseif S(i,3)==1 && S(j,3)==0
        if S(i,1)-S(j,1)>max_dx
            k=k+1;
            S(k,1)=(S(i,1)+S(j,1))/2;
            S(k,2)=0;
            S(k,5)=(S(i,9)+S(j,5))/2;
        end
    elseif S(i,3)==0 && S(j,3)==1
        if S(i,1)-S(j,1)>max_dx
            k=k+1;
            S(k,1)=(S(i,1)+S(j,1))/2;
            S(k,2)=0;
            S(k,5)=(S(i,5)+S(j,10))/2;
        end
    else
        if S(i,1)-S(j,1)>max_dx
            k=k+1;
            S(k,1)=(S(i,1)+S(j,1))/2;
            S(k,2)=0;
            S(k,5)=(S(i,9)+S(j,10))/2;
        end
    end
    %Gradients are too steep
    if S(i,3)==0 && S(j,3)==0
        if abs(S(i,5)-S(j,5))>max_df_Sc
            k=k+1;
            S(k,1)=(S(i,1)+S(j,1))/2;
            S(k,2)=0;
            S(k,5)=(S(i,5)+S(j,5))/2;
        end
    end
end
S=sortrows(S,-1);

%Aq series
k=size(A,1);
for i=1:size(A,1)-1;
    j=i+1;
    %Packets are just too far apart
    if A(i,3)==0 && A(j,3)==0
        if A(i,1)-A(j,1)>max_dx
            k=k+1;
            A(k,1)=(A(i,1)+A(j,1))/2;
            A(k,2)=1;
            A(k,6)=(A(i,6)+A(j,6))/2;
        end
    elseif A(i,3)==1 && A(j,3)==0
        if A(i,1)-A(j,1)>max_dx
```

```
                k=k+1;
                A(k,1)=(A(i,1)+A(j,1))/2;
                A(k,2)=1;
                A(k,6)=(A(i,13)+A(j,6))/2;
            end
        elseif A(i,3)==0 && A(j,3)==1
            if A(i,1)-A(j,1)>max_dx
                k=k+1;
                A(k,1)=(A(i,1)+A(j,1))/2;
                A(k,2)=1;
                A(k,6)=(A(i,6)+A(j,14))/2;
            end
        else
            if A(i,1)-A(j,1)>max_dx
                k=k+1;
                A(k,1)=(A(i,1)+A(j,1))/2;
                A(k,2)=1;
                A(k,6)=(A(i,13)+A(j,14))/2;
            end
        end

        %Slope is too great
        if A(i,3)==0 && A(j,3)==0
            if abs(A(i,6)-A(j,6))>max_df_Aq
                k=k+1;
                A(k,1)=(A(i,1)+A(j,1))/2;
                A(k,2)=1;
                A(k,6)=(A(i,6)+A(j,6))/2;
            end
        end
    end
    A=sortrows(A,-1);


    %% Step 7-Keep Aq and Sc in the same r domain, and remove
redundent particles

    %POPULATION CONTROL GOES HERE
    if size(S,1)>2
        S(:,end)=1;
        for i=2:size(S,1)-1
            % //Any deletion criterion can go here, just set
S(i,end)=0
            %Packets are just too close
            if abs(S(i,1)-S(i-1))<1e-6 && S(i,3)==0 && S(i-1,3)==0
                S(i,end)=0;
            end
            %Packets have nearly the same value
            if S(i,3)==0 && S(i-1,3)==0
               if abs(S(i,5)-S(i-1,5))<min_df
                   if abs(S(i,1)-S(i-1,1))<min_dx
                       S(i,end)=0;
                   end
               end
            end
```

```
        end
        S(~any(S(:,15),2),:)=[];
    end
    if size(A,1)>2
        A(:,end)=1;
        for i=2:size(A,1)-1
            %Get rid of NaNs
            if max(isnan(A(1,:)))==1
                A(i,end)=0;
            end
            %Delete shocks if Caq behind = Caq front
            if A(i,3)==1 && abs((A(i,9)*A(i,13))-
(A(i,10)*A(i,14)))<1e-6
                A(i,end)=0;
            end
            %Packets are just too close
            if abs(A(i,1)-A(i-1,1))< 1e-6 && A(i,3)==0 && A(i-1)==0
                A(i,end)=0;
            end
            %Packets have nearly the same value
            if A(i,3)==0 && A(i-1,3)==0
               if abs(A(i,6)-A(i-1,6))<min_df
                   if abs(A(i,1)-A(i-1,1))<min_dx
                       A(i,end)=0;
                   end
               end
            end
        end
        A(~any(A(:,15),2),:)=[];
    end


    %Keep S and A in same domain
    if abs(S(1,1)-A(1,1)) > min_dx
        if S(1,1)>A(1,1)        %S ahead, keepup A
            k=size(A,1)+1;
            A(k,1)=S(1,1)-(min_dx/2);
            A(k,2)=1;
            A(k,3)=0;
            A(k,4)=0;
            if S(1,3)==0;
                A(k,5)=S(1,5);
            else
                A(k,5)=S(1,9);
            end
            A=sortrows(A,-1);
        elseif A(1,1)>S(1,1)    %A ahead, keepup S
            k=size(S,1)+1;
            S(k,1)=A(1,1)-(min_dx/2);
            S(k,2)=0;
            S(k,3)=0;
            S(k,4)=0;
            if A(1,3)==0;
                S(k,6)=A(1,6);
            else
```

```
                S(k,6)=A(1,13);
            end
            S=sortrows(S,-1);
        end
    end

    %Put shock into Aq phase if it becomes the leading packet
    if A(1,1)>S(1,1) && A(1,3)==0 && t>PSM(1,1)
        A(1,3)=1;
        A(1,9)=S(1,5);
        A(1,10)=0;
        A(1,11)=A(1,7);
        A(1,12)=1;
        A(1,13)=A(1,6);
        A(1,14)=0;

A(1,4)=waq(A(1,9),A(1,10),A(1,11),A(1,12),A(1,13),A(1,14),phi);
    end


    %% STEP 8-Change the Pumping Fluid if needed
    if t>=T(Tindex+1,1)
        Tindex=Tindex+1;
        Sc=T(Tindex,2);

        if Sc>T(Tindex-1,2) %CO2 saturation is going up

            %Get rid of all fluid packets at the location r=r0, and
replace
            %with new Sc value
            if S(end,1)==r0
                k=size(S,1);
                S(k,1)=r0;
                S(k,5)=Sc;

S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));
            else
                k=size(S,1)+1;
                S(k,1)=r0;
                S(k,5)=Sc;

S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));
            end

        else
            %CO2 saturation is going down. Only replace the r0 spot
if you are
            %still above the minimum saturation, otherwise we must
keep the
            %saturation at the r0 spot, such that dissolution can
"eat away" at
            %the entrance node.
```

```
            k=size(S,1);
            if S(k,1)==r0
                if S(k,5) >= Sn
                    S(k,5)=Sc;

S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));
                end
            else
                k=k+1;
                S(k,1)=r0;
                S(k,5)=Sc;

S(k,4)=vsc(S(k,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(k,1),S
(k,8));
            end

            %deal with the last Aq node
            if A(end,1)<=r0
                A(end,1)=r0;
                A(end,6)=0;
            else
                k=size(A,1)+1;
                A(k,1)=r0;
                A(k,2)=1;
                A(k,6)=0;
            end
        end
    end


    %% STEP 9 -Interpolate Sc and Caq

    %Here we need to build the "True Value Data Set" for Sc and Caq
    e=1e-6;
    if size(S,1)>2 && size(A,1)>2
        %Build the True Set for Sc "Tsc"
        %Split S into normal and shocks
        S=sortrows(S,3);
        shockcount=sum(S(:,3));
        if shockcount>0                 %S had some shocks in it
            Tsc=zeros(size(S,1)+shockcount,3);
            Tsc(:,3)=1;
            k=size(S,1)-shockcount;
            Tsc(1:k,1)=S(1:k,1);
            Tsc(1:k,2)=S(1:k,5);
            % k=row in Tsc
            for j=size(S,1)-shockcount+1:size(S,1)
                k=k+1;
                Tsc(k,1)=S(j,1)+e;
                Tsc(k,2)=S(j,10);
                k=k+1;
                Tsc(k,1)=S(j,1)-e;
```

```
        Tsc(k,2)=S(j,9);
    end
else                                %S had no shocks in it
    Tsc=zeros(size(S,1),3);
    Tsc(:,1)=S(:,1);
    Tsc(:,2)=S(:,5);
    Tsc(:,3)=1;
end
S=sortrows(S,-1);
Tsc=sortrows(Tsc,-1);
if size(Tsc,1)>2
    for i=1:size(Tsc,1)-1
        if Tsc(i,1)==Tsc(i+1,1)
            Tsc(i,end)=0;
        end
    end
    Tsc(~any(Tsc(:,3),2),:)=[];
end

%Build the True Set for the Aq values "Taq"
%Split A into normal and shocks
A=sortrows(A,3);
shockcount=sum(A(:,3));
if shockcount>0                     %A had some shocks in it
    Taq=zeros(size(A,1)+shockcount,3);
    Taq(:,3)=1;
    k=size(A,1)-shockcount;
    Taq(1:k,1)=A(1:k,1);
    Taq(1:k,2)=A(1:k,6);
    %k=row in Taq
    for j=size(A,1)-shockcount+1:size(A,1)
        k=k+1;
        Taq(k,1)=A(j,1)+e;
        Taq(k,2)=A(j,14);
        k=k+1;
        Taq(k,1)=A(j,1)-e;
        Taq(k,2)=A(j,13);
    end
else                                %A had no shocks in it
    Taq=zeros(size(A,1),3);
    Taq(:,1)=A(:,1);
    Taq(:,2)=A(:,6);
    Taq(:,3)=1;
end
A=sortrows(A,-1);
Taq=sortrows(Taq,-1);
if size(Taq,1)>2
    for i=1:size(Taq,1)-1
        if Taq(i,1)==Taq(i+1,1)
            Taq(i,end)=0;
        end
    end
    Taq(~any(Taq(:,3),2),:)=[];
end
```

```
        %Now interpolate
        S(:,6)=interp1(Taq(:,1),Taq(:,2),S(:,1),'linear','extrap');
        A(:,5)=interp1(Tsc(:,1),Tsc(:,2),A(:,1),'linear','extrap');

        %Now check to make sure nothing interpolated "out of bounds"
        for i=1:size(S,1)
            if S(i,6)<0
                S(i,6)=0;
            elseif S(i,6)>Caqsat
                S(i,6)=Caqsat;
            end
        end

        for i=1:size(A,1)
            if A(i,5)<0
                A(i,5)=0;
            elseif A(i,5)>Sx
                A(i,5)=Sx;
            end
        end
    end


    %% STEP 10- Update the Integrals Column
    %There is no great way to run this in parallel. We have to
combine S
    %and A into one giant matrix and then tabulate the integral row
by row.

    %Build the giant P matrix
    P=[S;A];
    P=sortrows(P,-1);

    %Go through the P matrix "backwards" and tabulate the integrals
    for i=size(P,1)-1:-1:1
        dx=P(i,1)-P(i+1,1);
        if P(i+1,5)>0
            if P(i+1,3)==0;
                f1=P(i+1,1)*h*(1-P(i+1,5))*phi*(Caqsat-
P(i+1,6))/rhoc;
            else
                f1=P(i+1,1)*h*(1-P(i+1,10))*phi*(Caqsat-
P(i+1,14))/rhoc;
            end
        else
            f1=0;
        end
        if P(i,5)>0
            if P(i,3)==0;
                f2=P(i,1)*h*(1-P(i,5))*phi*(Caqsat-P(i,6))/rhoc;
            else
                f2=P(i,1)*h*(1-P(i,9))*phi*(Caqsat-P(i,13))/rhoc;
            end
        else
```

```
            f2=0;
        end
        P(i,8)=P(i+1,8)+(0.5*(f1+f2)*dx);
    end
    P(end,8)=0;

    %with the integrals tabulated, we need to split P back into S and
A
    P=sortrows(P,-2);
    Acount=sum(P(:,2));
    A=P(1:Acount,:);
    S=P(Acount+1:end,:);

    %resort A and S
    A=sortrows(A,-1);
    S=sortrows(S,-1);


    %% STEP 11-Update the Darcy, Characteristic and Shock Velocities

    %Normal Free-Phase CO2
    for i=1:size(S,1)
        if S(i,3)==0

S(i,7)=udc(S(i,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,S(i,1),S(i,8
));

S(i,4)=vsc(S(i,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,phi,Qtot,H,S(i,1),S
(i,8));
        end
    end

    %Normal Aq-Phase CO2
    for i=1:size(A,1)
        if A(i,3)==0

A(i,7)=udb(A(i,5),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,A(i,1),A(i,8
));
            A(i,4)=vaq(A(i,7),A(i,5),phi);
        end
    end


    %SHOCKUPDATE - Interpolate with nearest neighbors infront and
behind
    S=shockupdate_S(S);
    A=shockupdate_A(A);
    %Now check to make sure nothing interpolated "out of bounds"
    for i=1:size(S,1)
        if S(i,3)==1
            if S(i,9)<0;  S(i,9)=0; elseif S(i,9)>Sx;
S(i,9)=Sx;      end
            if S(i,10)<0; S(i,10)=0;elseif S(i,10)>Sx;
S(i,10)=Sx;     end
```

```
            if S(i,13)<0; S(i,13)=0;elseif S(i,13)>Caqsat;
S(i,13)=Caqsat; end
            if S(i,14)<0; S(i,14)=0;elseif S(i,14)>Caqsat;
S(i,14)=Caqsat; end
        end
    end
    for i=1:size(A,1)
        if A(i,3)==1
            if A(i,9)<0;  A(i,9)=0; elseif A(i,9)>Sx;
A(i,9)=Sx;      end
            if A(i,10)<0; A(i,10)=0;elseif A(i,10)>Sx;
A(i,10)=Sx;     end
            if A(i,13)<0; A(i,13)=0;elseif A(i,13)>Caqsat;
A(i,13)=Caqsat; end
            if A(i,14)<0; A(i,14)=0;elseif A(i,14)>Caqsat;
A(i,14)=Caqsat; end
        end
    end
    if S(1,1)>A(1,1) && S(1,3)==1; S(1,10)=0; S(1,14)=0; end
    if A(1,1)>S(1,1) && A(1,3)==1; A(1,10)=0; A(1,14)=0; end

    %Sc shocks
    for i=1:size(S,1)
        if S(i,3)==1

S(i,11)=udc(S(i,9),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,S(i,1),S(i,
8));

S(i,12)=udc(S(i,10),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,S(i,1),S(i
,8));
            S(i,4)=wsc(S(i,9),S(i,10),S(i,11),S(i,12),phi);
        end
    end
    %Aq shocks
    for i=1:size(A,1)
        if A(i,3)==1

A(i,11)=udb(A(i,9),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,A(i,1),A(i,
8));

A(i,12)=udb(A(i,10),Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,A(i,1),A(i
,8));

A(i,4)=waq(A(i,9),A(i,10),A(i,11),A(i,12),A(i,13),A(i,14),phi);
        end
    end

    %ERROR CHECKING SHOCKS IN S AND A
    if sum(S(:,3))>0    %Yep, S has some shocks in it
        S(:,end)=1;
        for i=1:size(S,1)
            if S(i,3)==1
                if abs(S(i,9)-S(i,10)) < 1e-6
                    S(i,end)=0;
                end
```

```
            end
            if max(isnan(S(i,:)))==1
                S(i,end)=0;
            end
        end
        S(~any(S(:,15),2),:)=[];
    end
    if sum(A(:,3))>0    %Yep, A has some shocks in it
        A(:,end)=1;
        for i=1:size(A,1)
            if A(i,3)==1
               if max(isnan(A(i,:)))==1
                   A(i,end)=0;
               end
            end
        end
        A(~any(A(:,15),2),:)=[];
    end


    %% STEP 12 - Outputs and Saving

    if show_outputs==1

        %Show the plot if needed
        if show_ScCaq_plot==1
            %if t>=frame_p*show_ScCaq_plot_interval

                Scp=zeros(1,2); Scs=zeros(1,2); i=1; j=1;
                for n=1:size(S,1)
                    if S(n,3)==0
                        Scp(i,1)=S(n,1); Scp(i,2)=S(n,5); i=i+1;
                    else
                        Scs(j,1)=S(n,1); Scs(j,2)=(S(n,9)+S(n,10))/2;
j=j+1;
                    end
                end

                Aqp=zeros(1,2); Aqs=zeros(1,2); i=1; j=1;
                for n=1:size(A,1)
                    if A(n,3)==0
                        Aqp(i,1)=A(n,1); Aqp(i,2)=A(n,6); i=i+1;
                    else
                        Aqs(j,1)=A(n,1);
Aqs(j,2)=(A(n,13)+A(n,14))/2; j=j+1;
                    end
                end
                Range=max(S(1,1),A(1,1));
                figure(1)

subplot(3,1,1),plot(Scp(:,1),Scp(:,2),'or',Scs(:,1),Scs(:,2),'ob')
                axis([r0 Range 0 1])
                ylabel('Free Phase S_c [-]')
                xlabel('r-location')
```

```
subplot(3,1,2),plot(Aqp(:,1),Aqp(:,2),'or',Aqs(:,1),Aqs(:,2),'ob')
                axis([r0 Range -Inf Inf])
                ylabel('C_a_q [kg/m^3]')
                xlabel('r-location')
                drawnow
                frame_p=frame_p+1;
            %end
        end


        %Show the mass error if needed
        if store_mass_error==1 || show_mass_error==1
           if t>=frame_me*mass_error_interval
                mass_co2_sc=0;
                mass_co2_aq=0;
                for i=1:size(S,1)-1
                    j=i+1;
                    R=0.5*(S(i,1)+S(j,1));
                    dR=S(i,1)-S(j,1);
                    if S(i,3)==0 && S(j,3)==0
                        dS=0.5*(S(i,5)+S(j,5));
                    elseif S(i,3)==0 && S(j,3)==1
                        dS=0.5*(S(i,5)+S(j,10));
                    elseif S(i,3)==1 && S(j,3)==0
                        dS=0.5*(S(i,9)+S(j,5));
                    else
                        dS=0.5*(S(i,9)+S(j,10));
                    end

mass_co2_sc=mass_co2_sc+(2*pi()*rhoc*phi*H*R*dS*dR);
                end
                for i=1:size(A,1)-1
                    j=i+1;
                    R=0.5*(A(i,1)+A(j,1));
                    dR=A(i,1)-A(j,1);
                    if A(i,3)==0
                        S1=(1-A(i,5));
                    else
                        S1=(1-A(i,9));
                    end
                    if A(j,3)==0
                        S2=(1-A(j,5));
                    else
                        S2=(1-A(j,10));
                    end
                    if A(i,3)==0
                        C1=A(i,6);
                    else
                        C1=A(i,13);
                    end
                    if A(j,3)==0
                        C2=A(j,6);
                    else
                        CS2=A(j,14);
```

```
                    end
                    C=(0.5)*(C1+C2);
                    Sb=(0.5)*(S1+S2);
                    mass_co2_aq=mass_co2_aq+(2*pi()*phi*H*R*dR*C*Sb);
                end
                out=[mass_co2_inj;mass_co2_sc;mass_co2_aq];
                percent_error=100*((mass_co2_aq+mass_co2_sc)-
mass_co2_inj)/mass_co2_inj;

                frame_me=frame_me+1;

                if store_mass_error==1
                    PE(frame_me,1)=t_days;
                    PE(frame_me,2)=percent_error;
                    PE(frame_me,3)=out(1,1);
                    PE(frame_me,4)=out(2,1);
                    PE(frame_me,5)=out(3,1);
                end

                if show_mass_error_plot==1
                    subplot(3,1,3),plot(PE(:,1),PE(:,2),'-r')
                    axis([0 Inf -3 3])
                    ylabel('Percent Error')
                    xlabel('time [days]')
                end
            end
        end

        clc
        dt
        t_days=t/24/3600
        if show_mass_error==1
            percent_error=PE(end,2)
        end
    end

    %Save the movie frame/workspace if needed
    if movie_on==1
        if t>=frame_m*movie_interval
            frame_m=frame_m+1;
            Movie_Frames{1,frame_m}=t;
            Movie_Frames{2,frame_m}=S;
            Movie_Frames{3,frame_m}=A;
            Movie_Frames{4,frame_m}=mass_co2_inj;
        end
    end

    if t>=frame_s*workspace_save_interval
        i=round(t/24/3600);
        fname = sprintf('%s_%i',filename,i);
        save(fname)
        clear Movie_Frames
        frame_s=frame_s+1;
    end
```

```
end

%% Calculate the Mass Balance
mass_co2_sc=0;
mass_co2_aq=0;
for i=1:size(S,1)-1
    j=i+1;
    R=0.5*(S(i,1)+S(j,1));
    dR=S(i,1)-S(j,1);
    if     S(i,3)==0&&S(j,3)==0; dS=0.5*(S(i,5)+S(j,5));
    elseif S(i,3)==0&&S(j,3)==1; dS=0.5*(S(i,5)+S(j,10));
    elseif S(i,3)==1&&S(j,3)==0; dS=0.5*(S(i,9)+S(j,5));
    else   S(i,3)==1&&S(j,3)==1; dS=0.5*(S(i,9)+S(j,10)); end
    mass_co2_sc=mass_co2_sc+(2*pi()*rhoc*phi*H*R*dS*dR);
end
for i=1:size(A,1)-1
    j=i+1;
    R=0.5*(A(i,1)+A(j,1));
    dR=A(i,1)-A(j,1);
    if A(i,3)==0; S1=(1-A(i,5)); else S1=(1-A(i,9)); end
    if A(j,3)==0; S2=(1-A(j,5)); else S2=(1-A(j,10)); end
    if A(i,3)==0; C1=A(i,6); else C1=A(i,13); end
    if A(j,3)==0; C2=A(j,6); else CS2=A(j,14); end
    C=(0.5)*(C1+C2);
    Sb=(0.5)*(S1+S2);
    mass_co2_aq=mass_co2_aq+(2*pi()*phi*H*R*dR*C*Sb);
end
out_label=[char('mass_co2_inj');char('mass_co2_sc
');char('mass_co2_aq ')];
out=[mass_co2_inj;mass_co2_sc;mass_co2_aq];
percent_error=100*((out(2,1)+out(3,1))-out(1,1))/out(1,1)

%% Output some results
Scp=zeros(1,2); Scs=zeros(1,2); i=1; j=1;
for n=1:size(S,1)
    if S(n,3)==0
        Scp(i,1)=S(n,1); Scp(i,2)=S(n,5); i=i+1;
    else
        Scs(j,1)=S(n,1); Scs(j,2)=(S(n,9)+S(n,10))/2; j=j+1;
    end
end

Aqp=zeros(1,2); Aqs=zeros(1,2); i=1; j=1;
for n=1:size(A,1)
    if A(n,3)==0
        Aqp(i,1)=A(n,1); Aqp(i,2)=A(n,6); i=i+1;
    else
        Aqs(j,1)=A(n,1); Aqs(j,2)=(A(n,13)+A(n,14))/2; j=j+1;
    end
end
Range=max(S(1,1),A(1,1));
figure(1)
subplot(3,1,1),plot(Scp(:,1),Scp(:,2),'or',Scs(:,1),Scs(:,2),'ob')
axis([r0 Range 0 1])
ylabel('Free Phase S_c [-]')
```

```
xlabel('r-location')
subplot(3,1,2),plot(Aqp(:,1),Aqp(:,2),'or',Aqs(:,1),Aqs(:,2),'ob')
axis([r0 Range 0 Caqsat])
ylabel('C_a_q [kg/m^3]')
xlabel('r-location')
drawnow

%%
mass_co2_sc=0;
mass_co2_aq=0;
for i=1:size(S,1)-1
    j=i+1;
    R=0.5*(S(i,1)+S(j,1));
    dR=S(i,1)-S(j,1);
    if     S(i,3)==0&&S(j,3)==0; dS=0.5*(S(i,5)+S(j,5));
    elseif S(i,3)==0&&S(j,3)==1; dS=0.5*(S(i,5)+S(j,10));
    elseif S(i,3)==1&&S(j,3)==0; dS=0.5*(S(i,9)+S(j,5));
    else   S(i,3)==1&&S(j,3)==1; dS=0.5*(S(i,9)+S(j,10)); end
    mass_co2_sc=mass_co2_sc+(2*pi()*rhoc*phi*H*R*dS*dR);
end
for i=1:size(A,1)-1
    j=i+1;
    R=0.5*(A(i,1)+A(j,1));
    dR=A(i,1)-A(j,1);
    if A(i,3)==0; CS1=A(i,6)*(1-A(i,5)); else CS1=A(i,13)*(1-A(i,9));
end
    if A(j,3)==0; CS2=A(j,6)*(1-A(j,5)); else CS2=A(j,14)*(1-
A(j,10)); end
    CS=(CS1+CS2)/2;
    mass_co2_aq=mass_co2_aq+(2*pi()*phi*H*R*CS*dR);
end
out=[mass_co2_inj;mass_co2_sc;mass_co2_aq];

clc
dt
t_days=t/24/3600
percent_error=100*((out(2,1)+out(3,1))-out(1,1))/out(1,1)
frame_me=frame_me+1;
PE(frame_me,1)=t_days;
PE(frame_me,2)=percent_error;
PE(frame_me,3)=out(1,1);
PE(frame_me,4)=out(2,1);
PE(frame_me,5)=out(3,1);
subplot(3,1,3),plot(PE(:,1),PE(:,2),'-r')
axis([0.25 Inf -3 3])
ylabel('Percent Error')
xlabel('time [days]')

%%
figure(2)
plot(Scp(:,1),Scp(:,2),'or',Scs(:,1),Scs(:,2),'^k')
hold on
plot(Aqp(:,1),Aqp(:,2)./Caqsat,'ob',Aqs(:,1),Aqs(:,2)./Caqsat,'^k')
axis([r0 Range 0 1])
ylabel('Free Phase S_c [-]')
```

```
xlabel('r-location')
drawnow


%%
i=round(t/24/3600);
fname = sprintf('%s_%i',filename,i);
save(fname)







function Q=shockupdate_A(A)

Aq=A;

%Update the Aqueous Phase Shocks
for i=1:size(Aq,1)
    if Aq(i,3)==1 %Shock
        %Update front of AQUEOUS SHOCK
        if i>3
            if Aq(i-1,3)==0 %Normal particle in the "x1" position
                if Aq(i-2,3)==0 %Normal particle in the "x2" position
                    dx21=Aq(i-2,1)-Aq(i-1,1);
                    dx10=Aq(i-1,1)-Aq(i,1);
                    %Update Sc
                    slope=(Aq(i-2,5)-Aq(i-1,5))/dx21;
                    Aq(i,10)=Aq(i-1,5)-(slope*dx10);
                    %Update Caq
                    slope=(Aq(i-2,6)-Aq(i-1,6))/dx21;
                    Aq(i,14)=Aq(i-1,6)-(slope*dx10);
                else %shock in the "x2" position
                    dx21=Aq(i-2,1)-Aq(i-1,1);
                    dx10=Aq(i-1,1)-Aq(i,1);
                    %Update Sc
                    slope=(Aq(i-2,9)-Aq(i-1,5))/dx21;
                    Aq(i,10)=Aq(i-1,5)-(slope*dx10);
                    %Update Caq
                    slope=(Aq(i-2,13)-Aq(i-1,6))/dx21;
                    Aq(i,14)=Aq(i-1,6)-(slope*dx10);
                end
            end
        end
        %Update back of FREE PHASE SHOCK
        if i<size(Aq,1)-2
            if Aq(i+1,3)==0 %Normal particle in the "x1" position
                if Aq(i+2,3)==0 %Normal particle in the "x2" position
                    dx21=Aq(i+1,1)-Aq(i+2,1);
                    dx10=Aq(i,1)-Aq(i+1,1);
                    %Update Sc
                    slope=(Aq(i+1,5)-Aq(i+2,5))/dx21;
```

```
                    Aq(i,9)=Aq(i+1,5)+(slope*dx10);
                    %Update Caq
                    slope=(Aq(i+1,6)-Aq(i+2,6))/dx21;
                    Aq(i,13)=Aq(i+1,6)+(slope*dx10);
                else %shock in the "x2" position
                    dx21=Aq(i+1,1)-Aq(i+2,1);
                    dx10=Aq(i,1)-Aq(i+1,1);
                    %Update Sc
                    slope=(Aq(i+1,5)-Aq(i+2,10))/dx21;
                    Aq(i,9)=Aq(i+1,5)+(slope*dx10);
                    %Update Caq
                    slope=(Aq(i+1,6)-Aq(i+2,14))/dx21;
                    Aq(i,13)=Aq(i+1,6)+(slope*dx10);
                end
            end
        end
    end
end

Q=Aq;
end
```

```
function Q=shockupdate_S(S)

Sc=S;

%Update the FREE Phase Shocks
for i=1:size(Sc,1)
    if Sc(i,3)==1 %Shock
        %In case the shock is the leading packet in the simulation
        if Sc(i,1)==S(1,1)
            Sc(i,10)=0;
        end
        %When the shock only has 1 nearest neighbor behind it
        if size(Sc,1)-i==1
            Sc(i,9)=Sc(end,5);
        end

        %Update front of FREE PHASE SHOCK
        if i>3
            if Sc(i-1,3)==0 %Normal particle in the "x1" position
                if Sc(i-2,3)==0 %Normal particle in the "x2" position
                    dx21=Sc(i-2,1)-Sc(i-1,1);
                    dx10=Sc(i-1,1)-Sc(i,1);
                    %Update Sc
                    slope=(Sc(i-2,5)-Sc(i-1,5))/dx21;
                    Sc(i,10)=Sc(i-1,5)-(slope*dx10);
                    %update Caq
                    slope=(Sc(i-2,6)-Sc(i-1,6))/dx21;
                    Sc(i,14)=Sc(i-1,6)-(slope*dx10);
                else %shock in the "x2" position
```

```
                    dx21=Sc(i-2,1)-Sc(i-1,1);
                    dx10=Sc(i-1,1)-Sc(i,1);
                    %Update Sc
                    slope=(Sc(i-2,9)-Sc(i-1,5))/dx21;
                    Sc(i,10)=Sc(i-1,5)-(slope*dx10);
                    %Update Caq
                    slope=(Sc(i-2,13)-Sc(i-1,6))/dx21;
                    Sc(i,14)=Sc(i-1,6)-(slope*dx10);
                end
            end
        end
        %Update back of FREE PHASE SHOCK
        if i<size(Sc,1)-2
            if Sc(i+1,3)==0 %Normal particle in the "x1" position
                if Sc(i+2,3)==0 %Normal particle in the "x2" position
                    dx21=Sc(i+1,1)-Sc(i+2,1);
                    dx10=Sc(i,1)-Sc(i+1,1);
                    %Update Sc
                    slope=(Sc(i+1,5)-Sc(i+2,5))/dx21;
                    Sc(i,9)=Sc(i+1,5)+(slope*dx10);
                    %Update Caq
                    slope=(Sc(i+1,6)-Sc(i+2,6))/dx21;
                    Sc(i,13)=Sc(i+1,6)+(slope*dx10);
                else %shock in the "x2" position
                    dx21=Sc(i+1,1)-Sc(i+2,1);
                    dx10=Sc(i,1)-Sc(i+1,1);
                    %Update Sc
                    slope=(Sc(i+1,5)-Sc(i+2,10))/dx21;
                    Sc(i,9)=Sc(i+1,5)+(slope*dx10);
                    %Update Caq
                    slope=(Sc(i+1,6)-Sc(i+2,14))/dx21;
                    Sc(i,13)=Sc(i+1,6)+(slope*dx10);
                end
            end
        end
    end
end

Q=Sc;
end




%Darcy Velocity of Brine
function u = udb(Sc,Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,r,Intg)

    if Sc<=Sn
        kc=0;
        kb=1;
    elseif Sc>=Sx
        kc=1;
        kb=0;
    else
```

```
        kc=(kcx)*(((Sc-Sn)/(Sx-Sn))^Coreyn);
        kb=(kbx)*(((Sx-Sc)/(Sx-Sn))^Coreym);
    end

    u=(1-(1/(1+(eta*(kb/kc)))))*((Qtot/(2*pi()*H*r))-((1/r)*Intg));
end




%Darcy Velocity of Free-Phase CO2
function u = udc(Sc,Sn,Sx,kcx,kbx,Coreyn,Coreym,eta,Qtot,H,r,Intg)

    if Sc<=Sn
        kc=0;
        kb=1;
    elseif Sc>=Sx
        kc=1;
        kb=0;
    else
        kc=(kcx)*(((Sc-Sn)/(Sx-Sn))^Coreyn);
        kb=(kbx)*(((Sx-Sc)/(Sx-Sn))^Coreym);
    end

    u=(1/(1+(eta*(kb/kc))))*((Qtot/(2*pi()*H*r))-((1/r)*Intg));
end




%Characteristic Velocity of the Aqueous-Phase CO2
function v = vaq(udb,Sc,phi)
    v = udb/(phi*(1-Sc));
end




%Characteristic Velocity of Free-Phase CO2
function v = vsc(Sc,Sn,Sx,kcx,kbx,n,m,eta,phi,Qtot,H,r,Int)

    E=eta;
    A=Qtot/(2*pi()*H*r) - (1/r)*Int;

    if Sc<=Sn
        v=0;
    elseif Sc>=Sx
        v=0;
    else
        v = -A*kcx*(-(Sc-Sn)/(-Sx+Sn))^n*E*kbx*((-Sx+Sc)/(-
Sx+Sn))^m*(m*Sc-m*Sn+n*Sx-n*Sc)/((-Sx+Sc)*(kcx*(-(Sc-Sn)/(-
Sx+Sn))^n+E*kbx*((-Sx+Sc)/(-Sx+Sn))^m)^2*(Sc-Sn));
```

```
    end

    v = v/phi;
end




%Shock velocity of the Aqueous-Phase CO2
function w = waq(Sc_be,Sc_in,udb_be,udb_in,Caq_be,Caq_in,phi)
    Sb_be=1-Sc_be;
    Sb_in=1-Sc_in;
    w = (1/phi)*((Caq_be*udb_be)-(Caq_in*udb_in))/((Caq_be*Sb_be)-
(Caq_in*Sb_in));
end




%Shock velocity of the Free-Phase CO2
function w = wsc(Sc_be,Sc_in,udc_be,udc_in,phi)
    w=(udc_be-udc_in)/(phi*(Sc_be-Sc_in));
end
```

Appendix 6

VI. Matlab codes to solve equation 72

```
% Z-direction 1D remobilization problem
% M01_Zdir_mainloop
% written by Erik J. Huber
% Last Edited 19 Jan 2015

% Description
% This program is designed to solve the 1D remobilized CO2 problem in
the
% nondimensional space subject to the choice of kc, kb, eta, Bo, and
% J-Leverett function. Given these parameters along with an initial
% condition of Sco(z*), the saturation profile will evolve until
reaching
% steady state, at which point the time will be recorded. This
program will
% also be set up to check for node resolution/time resolution, and
will
% enable the user to loop through a set of parameters.

% Governing Equations
% dS/dt* + d/dz*(U - (U/Bo)(dJ/dS)(dS/dz*))=0
% U = (kc*eta*f)/(1+(eta*f))
% eta = mu_c/mu_b
% f = kb/kc
% Bo = (g*delta(rho)*H)/(sigma*sqrt(phi/kappa))
% z* = z/H
% t* = t (kappa*delta(rho)*g)/(phi*mu_c*H)

clear all;
close all;
clc
%-------------------------------------------------------------------
------

% Main Loop Inputs
filename=('top_hat_testing');
Sco=0.25*1.2;          %Initial saturation in the z-direction,
Sc(z,t=0) [-]
Bo=0.20;                %Bond number [-]


% Relative permeability Inputs
Scmin=0.25;             %residual CO2 saturation [-]
Scmax=0.80;             %maximum CO2 saturation [-]
kcmax=0.5;              %maximum CO2 relative permeability [-]
kbmax=1;                %maximum brine relative permeability [-]
Coreyc=1;               %CO2 Corey exponent
Coreyb=2;               %brine Corey exponent
```

```
Dp=2.99;                %best-fit fractal dimension
np=0.94;                %best fit CO2 corey exponent for Brooks-
Corey-Hunt

% Other Inputs
phi=0.15;               %porosity of the reservoir [-]
eta=0.1;                %viscosity ratio [-]

% Code inputs
ndz=0.01;                   %spacing between nodes in solution
reltol_ss=1e-2;             %percent error to determine "L" in the
"true S.S. solution"
relative_error=1/exp(2);    %used to determine when solution reaches
steady state
efrequency=0.01;            %record/show things when "e" decreases by
this much
%-------------------------------------------------------------------
------

inputlist=[1,1.5,1.88,2,2.5,3,3.5,4;0.94,1.47,1.88,2.01,2.58,3.17,3.7
8,4.42]';
%inputlist=[4;4.42]';
Scf=Sco/Scmin;

Results=zeros(size(inputlist,1),8);
Results(:,1:2)=inputlist;
rrow=1;

%%
for rrow=4:4
    Coreyc=Results(rrow,1);     %CO2 Corey exponent
    np=Results(rrow,2);         %best fit CO2 corey exponent for
Brooks-Corey-Hunt

    for percolation=0:0
        % Run all the sub-functions
        S01_ReferenceTables
        S02_SSZ
        safety=4;
        attempts=0;
        while attempts < 8
            attempts=attempts+1;
            S03_Run
            if max(isnan(Z(:,2)))==0 %success
                break
            else                        %not a success, decrease the
timestep
                safety=2*safety;
            end
        end

        %Record the results from this run
        if percolation==0
            Results(rrow,3)=iterations;
```

```
            Results(rrow,4)=attempts;
            Results(rrow,5)=nt;
        else
            Results(rrow,6)=iterations;
            Results(rrow,7)=attempts;
            Results(rrow,8)=nt;
        end

        save(filename)
    end
end
```

```
% Z-direction 1D remobilization problem
% S01_Zdir_ReferenceTables
% written by Erik J. Huber
% Last Edited 26 May 2015

% Description
% This subroutine builds the appropriate reference tables for the
% Z-direction problem.

% Code outputs
% -------------
% Ref = matrix [Sc value , U , D]

% Governing Equations
% dS/dt* + d/dz*(U - D(dS/dz*))=0
% U = (kc*eta*f)/(1+(eta*f))
% D = (U/Bo)(dJ/dS)
% eta = mu_c/mu_b
% f = kb/kc

% Leverett's Function (from PetraSIM)
% J = aj1(1-S.) + aj2(1-S.)^2 + aj3(1-S.)^3
% S. = (Sb-Sbmin)/(1-Sbmin)
% aj1 = 1.417
% aj2 = -2.120
% aj3 = 1.263

%-------------------------------------------------------------------------
------

n=5001;                     %number of entries for this reference
table
ds=(Scmax-Scmin)/(n-1);     %gredations of CO2 saturation to build
the table

%First build the J(Sc) and dJ/dSc vectors
J=zeros(n,2);
J(:,1)=[Scmin:ds:Scmax]';
```

```
aj1=1.417;
aj2=-2.120;
aj3=1.263;
Sbmin=1-Scmax;
for i=1:size(J,1)
    Sb=1-J(i,1);
    Sstar=(Sb-Sbmin)/(1-Sbmin);
    J(i,2)=(aj1*(1-Sstar)) + (aj2*(1-Sstar)^2) + (aj3*(1-Sstar)^3);
end
for i=1:size(J,1)-1
    dJdSc(i,1)=(J(i,1)+J(i+1,1))/2;
    dJdSc(i,2)=(J(i+1,2)-J(i,2))/(J(i+1,1)-J(i,1));
end

% Build the reference tables
Ref=zeros(n,3);
for i=1:n
    %col 1: Sc
    S=Scmin + (i-1)*ds;
    Ref(i,1)=S;

    %col 2: U
    kb=kbmax*(((Scmax-S)/(Scmax-Scmin))^Coreyb);
    if percolation==0
        kc=kcmax*(((S-Scmin)/(Scmax-Scmin))^Coreyc);
    else
        Scx=Scmin+((1/phi)*((1.88*(1-phi))/((3/(3-Dp))-1.88)));
        Scp=(1/1.88)*(1.88*Scx-np*(Scx-Scmin));
        ap=kcmax*(((Scmax-Scmin)/(Scx-Scmin))^1.88)*(((Scx-
Scp)/(Scmax-Scp))^np);
        if S<=Scx
            kc=ap*(((S-Scmin)/(Scmax-Scmin))^1.88);
        else
            kc=kcmax*(((S-Scp)/(Scmax-Scp))^np);
        end
    end
    if kc==0
        f=0;
    else
        f=kb/kc;
    end
    U=kc*(f*eta/(1+(f*eta)));
    Ref(i,2)=U;

    %col 3: D

Ref(i,3)=(U/Bo)*interp1(dJdSc(:,1),dJdSc(:,2),S,'linear','extrap');
end
Ref2=Ref;
Ref=zeros(size(Ref2,1)+2,size(Ref2,2));
Ref(2:end-1,1:3)=Ref2(:,:);
Ref(end,1)=1;
clear Ref2
```

```
% Z-direction 1D remobilization problem
% S02_Zdir_SteadyStateSolution
% written by Erik J. Huber
% Last Edited 26 May 2015

% Description
% This subroutine builds the steady state solution for the problem.
This
% routine will use a search algorithm to hone in on the proper value
of L
% (the level location where the function changes from Scmin to
dS/dz). It
% will start with L=0.5, and then begin stepping with a half-the-
distance
% to the goal routine until a satisfactory result is reached

% The solution to this problem comes from solving 2 equations.
% dS/dz = Bo*(dJdS)^-1
% Sco = Scmin*L + integral(dS/dz from L to 1)

%-------------------------------------------------------------------
------

% The first step is to check if L=0 and there is no true "Scmin" zone
at
% the bottom, then the steady state solution will be found in a
different
% manner. To test this, (1) set L=0 (so z=0 is Scmin), (2) build a
steady
% state solution assuming this condition at z=0, (3) check the CO2
mass, if
% the CO2 mass is equal-you're done, if the CO2 is greater then you
need an
% Scmin area at the bottom, but if the mass is still less, then you
do not
% need any Scmin area at the bottom of the reservoir, in fact you
need to
% start z=0 with a larger Sc value than Scmin.



% Check L=0
L=0;
z=L;
S=Scmin;
dS=1e-5;
P=zeros(1,2);
P(1,1:2)=[z,S];
while P(end,1)<1
   S=S+dS;
   if S>Scmax
       S=Scmax;
```

```
    end
    i=size(P,1)+1;
    P(i,2)=S;
    dJ=interp1(dJdSc(:,1),dJdSc(:,2),S,'linear','extrap');
    dz=dS*dJ/Bo;
    P(i,1)=P(i-1,1)+dz;
end
%Calculate the CO2 mass given this new distribution
scvol=(Scmin*L)+trapz(P(:,1),P(:,2));

%%



% If scvol is greater than Sco, then you need an Scmin zone at the
bottom
% of the reservoir. Here is a searching algorithm to find the
appropriate L
% value to satisfy the proper mass balance
if scvol > Sco
    L=0.5;
    dL=0.5;
    stopflag=0;
    iterations=0;
    while stopflag==0
       %Build a matrix [z,Sc] starting at "L" and going to 1
       z=L;
       S=Scmin;
       dS=1e-5;
       P=zeros(1,2);
       P(1,1:2)=[z,S];
       while P(end,1)<1
           S=S+dS;
           if S>Scmax
               S=Scmax;
           end
           i=size(P,1)+1;
           P(i,2)=S;
           dJ=interp1(dJdSc(:,1),dJdSc(:,2),S,'linear','extrap');
           dz=dS*dJ/Bo;
           P(i,1)=P(i-1,1)+dz;
       end
       %Calculate the CO2 mass given this new distribution
       scvol=(Scmin*L)+trapz(P(:,1),P(:,2));
       %screen outputs
       clc
       steady_state_iterations=iterations
       %check to see if we can stop
       if 100*abs(scvol-Sco)/Sco < reltol_ss
           stopflag=1;
       else
           iterations=iterations+1;
           dL=dL/2;
           if scvol>Sco     %the volume was too high, so L must
increase
```

```
                L=L+dL;
            else              %the volume was too low, so L must
decrease
                L=L-dL;
            end
        end
        if iterations > 21
            stopflag=1;
        end
    end
    % At the end of the above loop, L should be the actual turning
point for
    % the function, or at least the most reasonable solution. We can
save the
    % entire profile as one matrix
    dz=1e-3;
    sszfine=zeros(1,2);
    sszfine(1,1:2)=[0,Scmin];
    z=0;
    passflag=0;
    while z<=1
        z=z+dz;
        if z<L
            sszfine(end+1,1)=z;
            sszfine(end,2)=Scmin;
        elseif z>=L && passflag==0
            sszfine(end+1,1)=L;
            sszfine(end,2)=Scmin;
            passflag=1;
            sszfine(end+1,1)=z;

sszfine(end,2)=interp1(P(:,1),P(:,2),z,'linear','extrap');
        else
            sszfine(end+1,1)=z;

sszfine(end,2)=interp1(P(:,1),P(:,2),z,'linear','extrap');
        end
    end
else




% If scvol is less than Sco, then there is NO SCMIN area at the
bottom of
% the reservoir, and instead you must increase the saturation at the
bottom
% of the reservoir, then see if the mass is okay
    L=(Scmax+Scmin)/2;
    dL=L;
    stopflag=0;
    iterations=0;
    while stopflag==0
```

```
        %Build a matrix [z,Sc] starting Sc at "L" at z=0
        z=0;
        S=L;
        dS=1e-5;
        P=zeros(1,2);
        P(1,1:2)=[z,S];
        while P(end,1)<1
            S=S+dS;
            if S>Scmax
                S=Scmax;
            end
            i=size(P,1)+1;
            P(i,2)=S;
            dJ=interp1(dJdSc(:,1),dJdSc(:,2),S,'linear','extrap');
            dz=dS*dJ/Bo;
            P(i,1)=P(i-1,1)+dz;
        end
        %Calculate the CO2 mass given this new distribution
        scvol=trapz(P(:,1),P(:,2));
        %screen outputs
        clc
        steady_state_iterations=iterations
        %check to see if we can stop
        if 100*abs(scvol-Sco)/Sco < reltol_ss
            stopflag=1;
        else
            iterations=iterations+1;
            dL=dL/2;
            if scvol>Sco     %the volume was too high, so the Sc at
z=0 must increase
                L=L-dL;
            else             %the volume was too low, so the Sc at z=0
must decrease
                L=L+dL;
            end
        end
        if iterations > 21
            stopflag=1;
        end
    end
    % At the end of the above loop, L should be the saturation at the
    % bottom of the domain required to match the mass
    dz=1e-3;
    sszfine=zeros(1,2);
    sszfine(1,1:2)=[0,L];
    z=0;
    while z<=1
        z=z+dz;
        sszfine(end+1,1)=z;
        sszfine(end,2)=interp1(P(:,1),P(:,2),z,'linear','extrap');
    end
end
```

```
% Z-direction 1D remobilization problem
% S03_Zdir_Run
% written by Erik J. Huber
% Last Edited 26 May 2015

% Description
% This subroutine updates the saturation profile until steady state
is
% reached

%-----------------------------------------------------------------
------

% Initialize Z matrix [z , Sc(z)]
Z=(ndz/2:ndz:1-(ndz/2))';
Z(1:25,2)=Sco;

% Build a low-resolution steady-state matrix to compare to Z during
the
% main loop
ssz=zeros(size(Z));
ssz(:,1)=Z(:,1);
ssz(:,2)=interp1(sszfine(:,1),sszfine(:,2),ssz(:,1),'linear');

% Estimate a maximum allowable time step for the solution to remain
stable
ndt=(ndz^2)/(safety*max(Ref(:,3)));

%%

clear Profiles
Profiles{1,1}=0;
Profiles{1,2}=Z;
Profilesframe=1;

Znew=Z;
C=ndt/ndz;
nt=0;
stopflag=0;
stamp=1;
while stopflag==0 && nt < 20
    nt=nt+ndt;

    %------------------
    % Update first node
    Sou=zeros(1,1);
    Uou=Sou;
    Dou=Sou;
    Jou=Sou;
    Sou(:,1)=(Z(2,2)+Z(1,2))/2;
    Uou=interp1(Ref(:,1),Ref(:,2),Sou(:,1),'linear','extrap');
    Dou=interp1(Ref(:,1),Ref(:,3),Sou(:,1),'linear','extrap');
    Jou=Uou-(Dou.*((Z(2,2)-Z(1,2))/ndz));
```

```
Jin=zeros(1,1);

Znew(1,2)=Z(1,2)-C.*(Jou(:,1)-Jin(:,1));

%----------------------
% Update interior nodes
Sou=zeros(size(Z,1)-2,1);
Uou=Sou;
Dou=Sou;
Jou=Sou;
Sou(:,1)=(Z(3:end,2)+Z(2:end-1,2))/2;
Uou=interp1(Ref(:,1),Ref(:,2),Sou(:,1),'linear','extrap');
Dou=interp1(Ref(:,1),Ref(:,3),Sou(:,1),'linear','extrap');
Jou=Uou-(Dou.*((Z(3:end,2)-Z(2:end-1,2))/ndz));

Sin=zeros(size(Z,1)-2,1);
Uin=Sin;
Din=Sin;
Jin=Sin;
Sin(:,1)=(Z(2:end-1,2)+Z(1:end-2,2))/2;
Uin=interp1(Ref(:,1),Ref(:,2),Sin(:,1),'linear','extrap');
Din=interp1(Ref(:,1),Ref(:,3),Sin(:,1),'linear','extrap');
Jin=Uin-(Din.*((Z(2:end-1,2)-Z(1:end-2,2))/ndz));

Znew(2:end-1,2)=Z(2:end-1,2)-C.*(Jou(:,1)-Jin(:,1));

%-----------------
% Update last node
Jou=zeros(1,1);

Sin=zeros(1,1);
Uin=Sin;
Din=Sin;
Jin=Sin;
Sin(:,1)=(Z(end,2)+Z(end-1,2))/2;
Uin=interp1(Ref(:,1),Ref(:,2),Sin(:,1),'linear','extrap');
Din=interp1(Ref(:,1),Ref(:,3),Sin(:,1),'linear','extrap');
Jin=Uin-(Din.*((Z(end,2)-Z(end-1,2))/ndz));

Znew(end,2)=Z(end,2)-C.*(Jou(:,1)-Jin(:,1));

%-------------------
%Set Z equal to Znew
Z=Znew;

%     plot(Z(:,2),Z(:,1))
%     drawnow

%Stop criteria would need to go here
e=sum(abs(ssz(:,2)-Z(:,2)))/sum(abs(ssz(:,2)-Sco));
if e < relative_error
    stopflag=1;
```

```
    end

    %Bad stop
    if max(isnan(Z(:,2)))==1
        break
    end

    %OUTPUTS and RECORDS
    if e < 1 - efrequency*stamp
        stamp=stamp+1;
        clc
        nt
        e
        Bo
        Scf
        safety

        k=size(Profiles,1)+1;
        Profiles{k,1}=nt;
        Profiles{k,2}=Z;
    end
end

k=size(Profiles,1)+1;
Profiles{k,1}=nt;
Profiles{k,2}=Z;
```

Appendix 7

VII. Derivation of nondimensional geometric parameters for the characterization of the base domain depicted in Figure 26d

Derivation of ℓ/W, g/W, η, and g/H for a single column of voids ($\omega = W$) assuming that the column consists of n equally spaced voids.

Global relationship

$$H = ng + n\ell$$

$$\frac{H}{W} = \frac{n}{W}(g + \ell)$$

$$\alpha = \frac{n\beta}{b}(g + \ell)$$

$$\alpha * \left(\frac{1}{g + \ell}\right) = \frac{n\beta}{b}$$

$$\alpha * \left(\frac{1}{g + \ell}\right) * \ell = \frac{n\beta}{b} * \ell$$

$$\alpha\lambda = n\beta\frac{1}{\mathcal{R}_e}$$

derive ℓ/W	derive g/W
$\frac{\ell}{W} = \frac{\ell}{b}\beta = \beta\mathcal{R}$	$\alpha = n\frac{g}{W} + n\frac{\ell}{W}$
$\frac{\ell}{W} = \frac{\beta\alpha\lambda}{n\beta}$	$\alpha = n\frac{g}{W} + n\frac{\alpha\lambda}{n}$
$\frac{\ell}{W} = \frac{\alpha\lambda}{n}$	$\frac{g}{W} = \frac{\alpha(1-\lambda)}{n}$

$$\eta = \frac{nb\ell}{HW} = n\left(\frac{b}{W}\right)\left(\frac{\ell}{W}\right)\left(\frac{1}{\alpha}\right) = n\beta\,\frac{\alpha\lambda}{n}\frac{1}{\alpha} = \beta\lambda$$

$$\eta = \beta\lambda$$

$$\frac{g}{H} = \frac{g}{\alpha W} = \frac{1-\lambda}{n}$$

Appendix 8

VIII. Vincent, et. al. Drying by Cavitation and Poroelastic Relaxations in Porous Media with Macroscopic Pores Connected by Nanoscale Throats

Drying by Cavitation and Poroelastic Relaxations in Porous Media with Macroscopic Pores Connected by Nanoscale Throats

Olivier Vincent,[1,*] David A. Sessoms,[1] Erik J. Huber,[2] Jules Guioth,[1] and Abraham D. Stroock[1,3,†]
[1]*School of Chemical and Biomolecular Engineering, Cornell University, Ithaca, New York 14853, USA*
[2]*Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, New York 14853, USA*
[3]*Kavli Institute at Cornell for Nanoscale Science, Cornell University, Ithaca, New York 14853, USA*
(Received 25 February 2014; revised manuscript received 22 May 2014; published 25 September 2014)

We investigate the drying dynamics of porous media with two pore diameters separated by several orders of magnitude. Nanometer-sized pores at the edge of our samples prevent air entry, while drying proceeds by heterogeneous nucleation of vapor bubbles—cavitation—in the liquid in micrometer-sized voids within the sample. We show that the dynamics of cavitation and drying are set by the interplay of the deterministic poroelastic mass transport in the porous medium and the stochastic nucleation process. Spatiotemporal patterns emerge in this unusual reaction-diffusion system, with temporal oscillations in the drying rate and variable roughness of the drying front.

DOI: 10.1103/PhysRevLett.113.134501 PACS numbers: 47.56.+r, 05.70.Ln, 47.55.dp, 89.75.-k

The desorption, or drying, of liquids from porous media is important in a variety of contexts, both in nature (e.g., the movement of water in plants [1]) and in technology (e.g., the synthesis [2] and characterization [3] of advanced materials). Studies of desorption have focused on the dynamics and pattern formation associated with drying [4–6] and the thermodynamics that define the shapes of desorption isotherms [7,8]. Despite this attention, uncertainties remain regarding the physical processes that govern desorption. It has been proposed that, rather than by receding of the liquid phase from the edges of the material, drying from porous media could occur by cavitation, i.e., the spontaneous formation of vapor bubbles either when the liquid tensile strength is attained [9] or by thermally activated nucleation in the metastable pore liquid [Fig. 1(a)] [10]. This process has been observed in simulations [8,11] and has been proposed on several occasions to explain the shape of desorption isotherms in nanoscale porous media [8,12,13] or the apparent emergence of drying events far from the evaporation front [14,15]. Yet, we are unaware of direct optical observation of desorption by cavitation or of an investigation of its effect on drying dynamics.

In this Letter, we present a tailored porous medium formed of microfabricated voids coupled to each other and the outside via a nanoporous substrate [Figs. 1(b) and 1(c)]. This extreme ink-bottle structure—large pore bodies connected via narrow throats [16]—has allowed us to observe the nucleation and growth of cavitation bubbles during drying [Fig. 1(d)] and to show that this process gives rise to interesting coupled drying dynamics that is tunable with geometry.

Drying occurs when a saturated porous medium is placed in a subsaturated atmosphere (relative humidity $p_v/p_{sat} < 1$, where p_v and p_{sat} are the vapor pressure and its saturation value, respectively). At high relative humidity, evaporation results in the formation of menisci in the pores at the surface of the material, until local mechanical and chemical equilibrium are established, which occurs when the liquid pressure P satisfies [17]

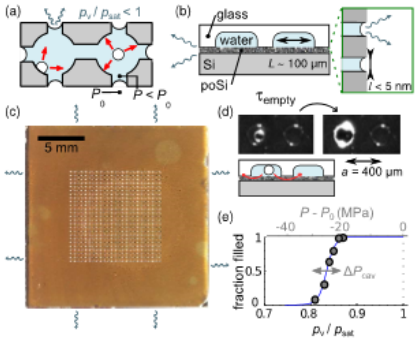


FIG. 1 (color online). (a) A porous medium with ink-bottle morphology may dry from inside by cavitation. (b) Schematic side view of our extreme ink-bottle geometry with microfabricated voids interconnected by porous silicon (poSi). (c) Photograph showing the top view of a sample with an array of 25 × 25 voids (bright dots) bonded to a layer of nanoporous silicon (orange background). Each void is dome shaped with a diameter of 200 μm and a depth of 27 μm. (d) Optical micrographs of a cavitation event (top view) and a schematic drawing of the water fluxes (side view). The left micrograph shows the bubble immediately after cavitation; the right micrograph shows the empty void. (e) Survival curve (see the text).

$$P - P_0 = \frac{RT}{v_m} \ln\left(\frac{p_v}{p_{\text{sat}}}\right) \quad (1)$$

and

$$P - P_0 > -\frac{2\sigma \cos\theta_r}{r_p}, \quad (2)$$

where $v_m = 1.805 \times 10^{-5}$ m^3/mol is the molar volume of the liquid, RT is the thermal energy, σ is the liquid surface tension, and θ_r is the receding contact angle of the meniscus in pores of radius r_p. At lower relative humidities, for which the pressure difference defined by (1) violates the inequality (2), the menisci will recede into the pores. This scenario is the classical process of drying by capillary invasion from the edges of the material [18].

Drying may also occur by a distinct mechanism: Equation (1) states that the pressure in the liquid phase is reduced relative to the standard pressure P_0, such that it is metastable with respect to the homogeneous or heterogeneous nucleation of bubbles of air and vapor [9,19] [Fig. 1(a)]. This mode of drying by cavitation may be favored in "ink-bottle" geometries in which small pores prevent invasion of the menisci and permit the development of large stresses within the pore liquid ["pore blocking" via Eq. (2)], while larger pores inside allow the nucleation of bubbles [Fig. 1(a)].

Here, we report on an extreme version of the ink-bottle geometry in which nanometer- and micrometer-sized voids coexist; previous studies of bidisperse porous media have been limited to small ($\lesssim$ twofold) differences in pore size [7,12,20]. Figures 1(b) and 1(c) present our system [21]: we formed discrete, micrometer-scale voids in the surface of glass via lithography; we formed a 5 μm-thick layer of interconnected, nanometer-scale pores (1–2 nm in radius) in the surface of silicon by anodization; and we bonded the glass and silicon anodically to couple the microvoids to each other and the outside environment via the nanopores.

We filled the evacuated samples by submerging them in liquid water at elevated pressure, then allowed them to dry in closed chambers with a controlled humidity, and followed the resulting dynamics by time-lapse photography. Equations (1) and (2) were always verified in our experiments, leaving cavitation as the only plausible mechanism for drying [21]. As seen in Fig. 1(d), we could follow discrete cavitation events within the microvoids during drying.

We first characterized the stability of the bulk liquid water within the microvoids by lowering the vapor pressure in step increments and counting the number of full voids after 1 day of equilibration [Fig. 1(e)]. Across samples, the shapes of the survival curves were similar, with probability 1/2 of cavitation for $p_v/p_{\text{sat}} = 0.83$–$0.86$, corresponding to liquid pressures in the range $P_{\text{cav}} = -20$ to -24 MPa from Eq. (1) (width $\Delta P_{\text{cav}} = 2$–$5$ MPa). This value is less negative than expected for homogeneous nucleation in water ($P_{\text{cav}} \sim -140$ MPa) [19], suggesting a heterogeneous process, but matches closely most measurements of P_{cav} in water [27]. The exact nucleation mechanism is, however, not important for the dynamics of drying.

Figure 2(a) presents a typical drying sequence for $p_v/p_{\text{sat}} = 0.84$ such that the boundary liquid pressure was $P_b \simeq -24$ MPa [Eq. (1)]. Drying proceeded in a mixed fashion, with both coherent, frontlike progression from the edges to the center and discrete events far away from the front. This observation suggests that the progression of cavitation was controlled by an interplay of deterministic mass transport and stochastic cavitation events that allow drying deep within the sample.

More surprisingly, despite the steady nature of the applied driving force, we found that cavitation and emptying of the voids proceeded in a punctuated manner, with bursts of one to seven events separated by periods with no events [Fig. 2(b)]. This progression of bursts persisted until all voids were empty [21] and showed long time correlations [Fig. 2(c)]. This emergent behavior suggests links to the dynamics in other out-of-equilibrium systems [28] such as the periodic relaxations in a continuously driven sand pile [29], which motivates the search for mechanisms of positive and negative feedback.

We considered but could exclude positive feedback based on acoustic emissions from cavitation events [21]. The period between successive bursts [$\simeq$15 min; see Figs. 2(b) and 2(c)] was similar to the time scale for bubble growth after cavitation [$\tau_{\text{empty}} \simeq 3.5$ min [21]; see Fig. 1(d)], which suggests that the bursting behavior arose from a hydraulic coupling due to the release of water into the surrounding medium during emptying.

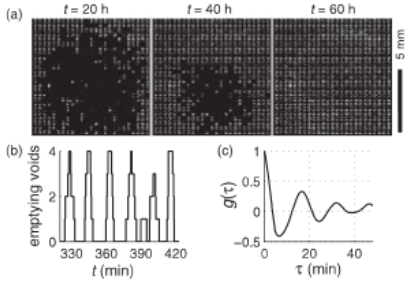


FIG. 2. Drying dynamics. (a) Optical micrographs showing state of voids in a sample as in Fig. 1(c) (only the array of voids was imaged) during drying at $p_v/p_{\text{sat}} = 0.84$. Empty voids appear bright. Evaporation occurred from all four edges of the samples. (b) Number of voids in the process of emptying as a function of time. (c) Autocorrelation of the time series as in (b) for all 625 cavitation events.

In order to check this hypothesis, we developed a model coupling deterministic mass transport and stochastic cavitation kinetics. We treat fluid transport within the porous silicon as poroelastic diffusion [30]:

$$\frac{\partial P}{\partial t} = C\nabla^2 P, \tag{3}$$

where $C = \kappa K_{\mathrm{liq}}/\phi = 5 \times 10^{-8}$ m^2/s is the poroelastic diffusivity, $\kappa = 1.44 \times 10^{-17}$ m^2/(Pa.s) is the Darcy permeability of porous silicon, $\phi = 0.6$ is its porosity, and $K_{\mathrm{liq}} = 2.2$ GPa is the bulk modulus of liquid water.

We describe cavitation kinetics in each void by the classical nucleation theory [19,27], leading to the following expression for the rate of nucleation:

$$k_{\mathrm{cav}}(t) = \Gamma_0 V \exp\left(-\frac{E_{\mathrm{nucl}}(P)}{k_B T}\right), \tag{4}$$

where Γ_0 is a kinetic prefactor, V is the void volume, and k_B is Boltzmann's constant. In the energy barrier for nucleation, $E_{\mathrm{nucl}} = 16\pi\sigma^3/[3(P - p_{\mathrm{sat}})^2]$, $\sigma \approx 0.02$ N/m is an effective surface tension with a value typical of those used to match the observed kinetics of cavitation of water in a variety of experiments [17,27,31]. In order to account for spatial heterogeneities, we also allowed σ to vary with a narrow distribution ($\Delta\sigma/\sigma \sim 0.02$–$0.07$). We adjusted σ and $\Delta\sigma$ to fit the experimental survival curves [Fig. 1(e), blue line].

We numerically solved Eq. (3) with an explicit finite difference scheme [32] for the evolution of the pressure field, $P(x, y, t)$. The four edges of the sample were maintained at a fixed liquid pressure P_b set by the outside humidity [Eq. (1)]. At each time step Δt, the probability of nucleation $k_{\mathrm{cav}} \times \Delta t$ was calculated in each void [Eq. (4)], and cavitation was triggered with a Monte Carlo scheme. If cavitation occurred, the pressure in the void was set to $P = p_{\mathrm{sat}}$ (liquid-vapor equilibrium) until the void was empty of liquid. After emptying, the vapor pressure in the void was allowed to evolve in local equilibrium [Eq. (1)] with the liquid pressure P in the neighboring region.

The sequence in Figs. 3(a)—3(d) presents snapshots of the calculated pressure field during a burst of four cavitation events under conditions matching the experiment shown in Fig. 2 [33]. When a void cavitates, its pressure jumps from the negative value in the surrounding matrix to p_{sat} [red pixels in Fig. 3(a)]. This transition generates outward flow, emptying of the void, and the growth of a resaturated zone in the surrounding matrix [Fig. 3(b), white lines]. Outside of this zone, strong metastability persists and further cavitation events can occur [Fig. 3(b), events 1 and 2]; inside this zone, metastability is reduced and the nucleation rate is negligible. The resaturated zones of multiple events grow and merge [Fig. 3(c)] until further cavitation is suppressed throughout the sample [Fig. 3(d)]. Once the cavitated voids have emptied, they cease to act as sources for the rehydration of their surroundings; the pressure in the sample begins to decrease again due to continued evaporation at the boundary, and a new cycle of cavitation (that is, a burst) can start.

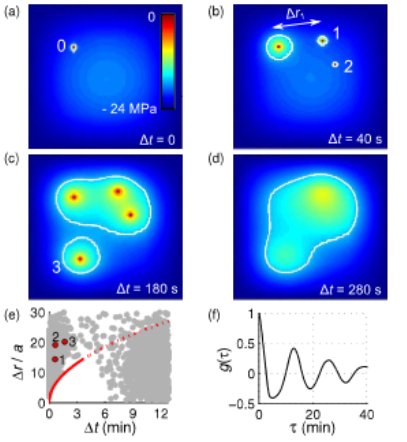


FIG. 3 (color online). Time evolution of the simulated pressure field in a 25 × 25 voids sample as in Fig. 2(a). (a) First cavitation event within a burst, labeled 0. (b) Growing exclusion zones (tracked by white contour lines at −17 MPa). (c) Exclusion zones merge. (d) Retraction of the exclusion zones. (e) Spatiotemporal representation of the drying dynamics for the full simulation, showing spatial (Δr) versus temporal (Δt) separations for all pairs of cavitation events. Events 1–3 from (a)—(d) (with event 0 at origin) are represented as red dots. The red line (continuous for $\Delta t < \tau_{\mathrm{empty}}$ and dashed afterwards) corresponds to $\Delta r = \alpha\sqrt{C\Delta t}$ with $\alpha = 1.7$. (f) Autocorrelation function of all emptying events in frame (e).

Figure 3(e) presents the spatiotemporal statistics of all cavitation events during the simulation. A pair of cavitation events separated by a time Δt can exist only outside of an exclusion zone (red line), the spatial extent of which grows as $\Delta r_{\mathrm{excl}} \sim \sqrt{C\Delta t}$ during a time scale on the order of the average emptying time τ_{empt}. The growth in $t^{1/2}$ arises from poroelastic diffusion [Eq. (3)].

The similarity of the autocorrelation function (which measures the bursting behavior) of simulated events [Fig. 3(f)] to that observed experimentally [Fig. 2(c)] supports the conclusion that the time-dependent response emerges due to this *suppressive* effect based on hydraulic coupling. As will be shown below, this coupling can be tuned experimentally by varying the geometry of the ink-bottle system.

The emergence of this complex response from the coupling of diffusion [Eq. (3)] and nonlinear reaction

[Eq. (4)] points to a particular analogy with the relaxation of metastable states in reaction-diffusion dynamics that leads, for example, to pattern formation during the precipitation of supersaturated solutions [34].

We turn to the identification of grouped parameters that control spatiotemporal drying patterns in these ink-bottle materials, with pressure diffusion and cavitation kinetics as the basic ingredients. First, we define a nondimensional interaction between voids to capture the negative feedback illustrated in Fig. 3. If the spatial range of the poroelastic coupling ($\Delta r_{\mathrm{excl}} \sim \sqrt{C \times \tau_{\mathrm{empt}}}$) is smaller than the distance a between voids, the interaction should vanish. We thus define a nondimensional parameter comparing these two length scales:

$$\zeta = \tau_{\mathrm{empt}} \times C/a^2, \tag{5}$$

which is also equal to the ratio of emptying time to the time $\tau_{\mathrm{diff}} = a^2/C$ for pressure information to travel from one void to its neighbor.

Second, we define a nondimensional driving force for cavitation:

$$\delta = (P_{\mathrm{cav}} - P_b)/\Delta P_{\mathrm{cav}}, \tag{6}$$

where P_b is the liquid pressure at the boundary defined by the external humidity [Eq. (1)] and P_{cav} and ΔP_{cav} parametrize the survival curve [Fig. 1(e)]. We expect δ to be a natural measure of the roughness of the drying front [21]: for strong driving forces ($\delta \gg 1$), we expect sharp fronts, controlled by the diffusion of the boundary pressure into the sample; for weak driving forces (small positive and negative values of δ), we expect spatially random events controlled by the kinetics of cavitation.

We investigated experimentally the effect of the two dimensionless parameters established above. First, the coupling parameter ζ [Eq. (5)] was varied by using samples with the same intervoid spacing, $a = 400\ \mu$m, but with a void volume that was ~200-fold smaller than that of the samples discussed up to this point. This geometrical change lowered the interaction parameter from $\zeta = 70$ (Fig. 4, top row) to $\zeta = 0.1$ (Fig. 4, bottom row) by decreasing the emptying time by a factor of 700. As predicted, the negative feedback between cavitation events was dramatically reduced, as can be seen in the nearly complete disappearance of the exclusion zones for both the strong and low driving force cases (Fig. 4, bottom). Furthermore, the time sequence of cavitation showed no measurable temporal correlations.

Second, to investigate the impact of driving force δ, we exposed the same samples to different humidities (Fig. 4; δ increases from left to right). As predicted, we observed that the drying front became much sharper for higher values of δ, with little effect on the hydraulic coupling (as seen in spatiotemporal maps). This evolution of the front shape is similar to that reported for evaporative drying of porous media [4,35]. The important distinction here is that drying

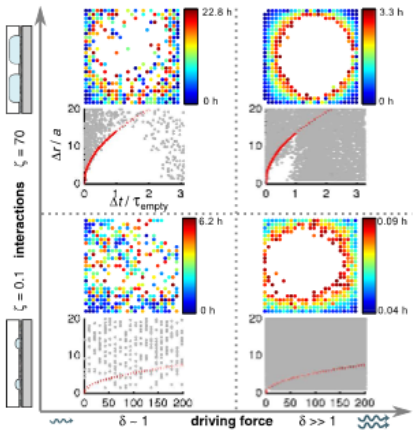


FIG. 4 (color online). Experimental state diagram of drying dynamics as a function of driving force and interactions between voids. The upper diagram in each quadrant is a spatial representation of the drying front when half of the voids had cavitated (color indicates event time). The lower diagram is the measure of temporal correlations introduced in Fig. 3(e) (the different data point densities reflect the impact of geometry or driving force on drying speed). The top left experiment is the one reported in Fig. 2.

occurs via cavitation at localized, spatially random points beyond the drying front, without percolation of a vapor path to the surface. Our experiments thus provide the first direct, experimental support of the proposal that this evolution of the pattern of drying with the driving force may result from cavitation [14,15].

Finally, we note that our simulations based on Eqs. (3) and (4) reproduce the state diagram in Fig. 4 well [21], providing further support for the relevance of our model.

In conclusion, we have used extreme ink-bottle geometries to provide direct optical observation of drying of a porous medium by cavitation. We have further elucidated aspects of the emergent dynamics of this process and shown that we can explain them quantitatively with a model that couples poroelastic transport and cavitation kinetics. This mechanism represents a new route to a nonlinear reaction-diffusion process, with mechanical rather than molecular diffusion and with nonlinear kinetics coming from nucleation rather than from reaction. We expect that similar coupled dynamics should occur in highly heterogeneous porous media, such as the ones found in rocks or plant tissues. In the geological context, the propagation of drying stresses could lead to patterns of nucleation of gases from supersaturated solutions that are formed, for example, during the sequestration of carbon dioxide [36]. In plant

tissues, the hydraulic interactions upon drying and cavitation might play a role in the physiological response to drought, as suggested by early results from Dixon, Grace, and Tyree [37]. We expect that microfabricated structures of the type presented here will provide useful avenues for the exploration of these and other phenomena.

The authors thank Eugene Choi for the measurement of nitrogen isotherms. This work was supported by the National Science Foundation (CBET-0747993 and CHE-0924463), the Air Force Office of Scientific Research (FA9550-09-1-0188), and the Camille Dreyfus Teacher-Scholar Awards program and was performed in part at the Cornell NanoScale Facility, a member of the National Nanotechnology Infrastructure Network (National Science Foundation; Grant No. ECCS-0335765).

*orv3@cornell.edu
†ads10@cornell.edu

[1] A. D. Stroock, V. V. Pagay, M. A. Zwieniecki, and N. M. Holbrook, Annu. Rev. Fluid Mech. **46**, 615 (2014).
[2] C. J. Brinker and G. W. Scherer, *Sol-Gel Science* (Academic, New York, 1990).
[3] E. P. Barrett, L. G. Joyner, and P. P. Halenda, J. Am. Chem. Soc. **73**, 373 (1951).
[4] T. M. Shaw, Phys. Rev. Lett. **59**, 1671 (1987).
[5] M. Prat, Chem. Eng. J. (Lausanne) **86**, 153 (2002).
[6] P. Lehmann, S. Assouline, and D. Or, Phys. Rev. E **77**, 056309 (2008).
[7] D. Wallacher, N. Künzner, D. Kovalev, N. Korr, and K. Knorr, Phys. Rev. Lett. **92**, 195704 (2004).
[8] P. I. Ravikovitch and A. V. Neimark, Langmuir **18**, 9830 (2002).
[9] R. K. Schofield, Discuss. Faraday Soc. **3**, 105 (1948).
[10] D. Or and M. Tuller, Water Resour. Res. **38**, 19 (2002).
[11] L. Sarkisov and P. A. Monson, Langmuir **17**, 7600 (2001).
[12] A. Grosman and C. Ortega, Langmuir **27**, 2364 (2011).
[13] F. Bonnet, M. Melich, L. Puech, and P. E. Wolf, Europhys. Lett. **101**, 16010 (2013).
[14] A. Sarkar, S. Chaudhuri, S. Wang, F. Kirkbir, and H. Murata, J. Sol-Gel Sci. Technol. **2**, 865 (1994).
[15] G. W. Scherer and D. M. Smith, J. Non-Cryst. Solids **189**, 197 (1995).
[16] L. H. Cohan, J. Am. Chem. Soc. **66**, 98 (1944).
[17] T. D. Wheeler and A. D. Stroock, Nature (London) **455**, 208 (2008).
[18] G. W. Scherer, J. Am. Ceram. Soc. **73**, 3 (1990).
[19] P. G. Debenedetti, in *Metastable Liquids: Concepts and Principles*, edited by J. M. Prausnitz and L. Brewer (Princeton University Press, Princeton, NJ, 1996).
[20] L. Bruschi, G. Mistura, L. Liu, W. Lee, U. Gösele, and B. Coasne, Langmuir **26**, 11894 (2010).
[21] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.113.134501, which includes Refs. [22–26], for details on experiments, model, simulations, and expanded discussion.
[22] J. Chirife and S. L. Resnik, J. Food. Sci. **49**, 1486 (1984).
[23] A. Rogacs, J. E. Steinbrenner, J. A. Rowlette, J. M. Weisse, X. L. Zheng, and K. E. Goodson, J. Colloid Interface Sci. **349**, 354 (2010).
[24] F. Caupin and A. D. Stroock, *The Stability Limit and Other Open Questions on Water at Negative Pressure*, Liquid Polymorphism, edited by H. E. Stanley (John Wiley & Sons, Inc., Hoboken, New Jersey, 2013) pp. 51–80.
[25] T. D. Wheeler and A. D. Stroock, Langmuir **25**, 7609 (2009).
[26] J. Bear, *Dynamics of Fluids in Porous Media* (Courier Dover Publications, Mineola, New York, 1988).
[27] F. Caupin and E. Herbert, C.R. Phys. **7**, 1000 (2006).
[28] M. C. Cross and P. C. Hohenberg, Rev. Mod. Phys. **65**, 851 (1993).
[29] H. M. Jaeger, C.-h. Liu, and S. R. Nagel, Phys. Rev. Lett. **62**, 40 (1989).
[30] M. A. Biot, J. Appl. Phys. **12**, 155 (1941).
[31] K. Davitt, A. Arvengas, and F. Caupin, Europhys. Lett. **90**, 16002 (2010).
[32] J. Crank, *The Mathematics of Diffusion* (Oxford University Press, New York, 1979).
[33] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.113.134501 for the video corresponding to the simulation on Fig. 3.
[34] K. H. Stern, Chem. Rev. **54**, 79 (1954).
[35] M. Prat and F. Bouleux, Phys. Rev. E **60**, 5647 (1999).
[36] H. E. Huppert and J. A. Neufeld, Annu. Rev. Fluid Mech. **46**, 255 (2014).
[37] M. A. Dixon, J. Grace, and M. T. Tyree, Plant Cell Environ. **7**, 615 (1984).

Appendix 9

IX. Derivation of the Maxwell elliptic effective medium permeability defined in equation 128.

Derivation of keff following the work of Zimmerman (1996)

Let N be the number of ellipses in a given region
Let $r = k_1/k_0$ the ratio of permeability of the inclusions to the background matrix
Let $\mathcal{R}_e$ be the aspect ratio of the ellipse, minor axis:major axis
Let a be half the length of the major axis

The solution to the inner velocity potential is given by (1a)

$$Q_x + iQ_y \approx Q_0\left[1 - \frac{Na^2\mathcal{R}_e(1+\mathcal{R}_e)(1-r)}{2(1+\mathcal{R}_e r)(\mathcal{R}_e + r)}\frac{\langle[(\mathcal{R}_e + r)\cos\delta - i(1+\mathcal{R}_e r)\sin\delta]e^{i\delta}\rangle}{\bar{z}^2}\right]$$

In our problem, the flow field is always oriented along $\delta = 0$, therefore

$$\langle[(\mathcal{R}_e + r)\cos\delta - i(1+\mathcal{R}_e r)\sin\delta]e^{i\delta}\rangle = (\mathcal{R}_e + r)$$

And equation (1a) becomes (1b)

$$Q_x + iQ_y \approx Q_0\left[1 - \frac{Na^2\mathcal{R}_e(1+\mathcal{R}_e)(1-r)}{2(1+\mathcal{R}_e r)}\frac{1}{\bar{z}^2}\right]$$

The solution to the outer velocity potential is given by (2)

$$Q_x + iQ_y = Q_0\left[1 - \frac{(k_0 - k_\epsilon)R^2}{(k_0 + k_\epsilon)\bar{z}^2}\right]$$

Setting (1b)=(2)

$$\frac{(k_0 - k_\epsilon)}{(k_0 + k_\epsilon)} = \frac{(1+\mathcal{R}_e)(1-r)}{2(1+\mathcal{R}_e r)}\frac{Na^2\mathcal{R}_e}{R^2}$$

In this context, the right most fraction is the area fraction of ellipses within the larger outer region of size R^2.
Let

$$\eta = \frac{Na^2\mathcal{R}_e}{R^2}$$

$$G = \frac{(1+\mathcal{R}_e)(1-r)}{2(1+\mathcal{R}_e r)}$$

$$\frac{k_0 - k_\epsilon}{k_0 + k_\epsilon} = G\eta$$

$$k_0 - k_\epsilon = G\eta k_0 + G\eta k_\epsilon$$

$$k_0(1 - G\eta) = k_\epsilon(1 + G\eta)$$

$$\frac{k_\epsilon}{k_0} = \frac{1 - G\eta}{1 + G\eta}$$

For the special case of $r \to \infty$

$$\lim_{r\to\infty} G = \frac{(1+\mathcal{R}_e)(1-r)}{2(1+\mathcal{R}_e r)} = \frac{-(1+\mathcal{R}_e)}{2\mathcal{R}_e}$$

Appendix 10

X. Derivation of the parallel path effective medium model defined in equation 134.

The total resistance along the non-void path

$$R_0 = \frac{L}{kA} = \frac{\alpha W}{k_0(W-b)} = \frac{\alpha}{k_0(1-\beta)}$$

The total resistance along the void path

$$R_{01} = R_0 + R_1 = \frac{ng}{k_0 b} + \frac{n\ell}{k_1 b}$$

$$R_{01} = \frac{n}{\frac{k_0 b}{W}} * \frac{g}{W} + \frac{n}{\frac{k_1 b}{W}} * \frac{\ell}{W} = \left(\frac{n}{k_0\beta} * \frac{\alpha(1-\lambda)}{n}\right) + \left(\frac{n}{k_1\beta} * \frac{\alpha\lambda}{n}\right)$$

$$R_{01} = \left(\frac{\alpha(1-\lambda)}{k_0\beta}\right) + \left(\frac{\alpha\lambda}{k_1\beta}\right) = \left(\frac{\alpha k_1(1-\lambda) + \alpha k_0\lambda}{k_0 k_1\beta}\right)$$

R0 and R01 are in parallel, combine into total resistance

$$\frac{1}{R_{TOT}} = \frac{1}{R_0} + \frac{1}{R_{01}} = \frac{k_0(1-\beta)}{\alpha} + \frac{k_0 k_1\beta}{\alpha(k_1(1-\lambda) + k_0\lambda)}$$

Now consider the general equation

$$Q = \frac{1}{R_{TOT}}\Delta P$$

$$u = \kappa_e * \frac{\Delta P}{H}$$

$$u * W = \kappa_e * \frac{\Delta P}{\alpha * W} * W$$

$$Q = \kappa_e * \frac{\Delta P}{\alpha}$$

$$\kappa_\epsilon = k_0(1-\beta) + \frac{k_0 k_1 \beta}{(k_1(1-\lambda) + k_0\lambda)}$$

Divide each side by k_0

$$k_{eff} = (1-\beta) + \frac{k_1 \beta}{(k_1(1-\lambda) + k_0\lambda)}$$

On the right hand term, divide numerator and denominator by kappa_1

$$k_{eff} = (1-\beta) + \frac{\beta}{(1-\lambda) + \frac{k_0}{k_1}\lambda}$$

$$\lim_{\kappa_1 \to \infty} k_{eff} = (1-\beta) + \frac{\beta}{(1-\lambda)}$$

Appendix 11

XI. Matlab code to solve for cavitation dynamics in the microscale model

```
% A01_TransientFiberMatrix_Inputs
% Last Edited 20 July 2015


%-------------------------------------------------------------------
------
% Description
% This code package will build an ordered matrix of fibers and then
solve
% the transient pressure field, including cavitation dynamics. An
adaptive
% mesh is used to capture any geometry input by the user.

% Revision History
% ----------------
% v01 - created

clear all; close all; clc
%-------------------------------------------------------------------
------
% Inputs
filename=('test');

numv=10;         %number of voids within a single column of the domain
numvx=10;        %number of columns in the total domain
alpha=1;        %aspect ratio of the entire domain
beta=0.50;       %projected void area to domain width in single column
lambda=1;        %nanopore path length versus void path length
minnodes=3;      %minimum number of nodes defining the smallest
distance
W=200e-6;        %width of the "unit cell" [m]
Hfiber=20e-9;   %height of the fibers [m]


% Properties regarding the physical sample and the connected fibers
kappa=1.90e-17;                     % permeability of the porous
silica [m^2 Pa^-1 s^-1]
porosity=0.5;                       % porosity of the porous media [-
]
porous_depth=5e-6;                  % depth of domain [m]
emptyclusters_are_shortcircuits=0;  % 1=Yes 0=No
% if you set "no" to the variable above, what should the new
permeability be
kappa_new=1.90e-17;                 % new permeability of the porous
silica after empty [m^2 Pa^-1 s^-1]
```

```
% Properties regarding pressure and cavitation dynamics
P_initial=0;                        % Initial pressure in the domain
[Pa]
P_top= -45e6;                       % Pressure at the drying side
[Pa]
P_bottom= 0;                        % Pressure at the liquid filled
side [Pa]
mean_sigma = 0.019;                 % average surface tension inside
the voids
stdev_sigma = 0.025*mean_sigma;


% Parameters controlling plotting, saving and stopping criteria
showscreenoutputs=1;                % show cavitation count and
simulation time on the main screen, 1=Yes 0=No
showoutputsfrequency=1e-3;          % show screen outputs every
______ seconds
showpatternfigure=1;                % show the cluster pattern
generated, 1=Yes 0=No

showpressurefigure=0;               % show the dynamic pressure field
during run, 1=Yes, 0=No
showpressurefrequency=1e-3;         % show the dynamic pressure field
during run every ____ seconds
showpressureDtype=3;                % show the dynamic pressure field
in: 2=2D 3=3D

finalruntime=100;                   % set this to -1 if you want no
stopping time, run will end when all clusters have cavitated, be
careful!!!
savepressurefield=1;                % save the pressure field for
movie purposes, 1=Yes 0=No
savepressurefrequency=1e-1;         % save the pressure field every
_____ seconds
savefilefrequency=1e3;              % save the entire workspace every
_____ seconds

recordglobalflux=1;                 % calculate and save the global
flux, 1=Yes 0=No
fluxfrequency=1e-1;                 % calculate and record the global
flux every _________ seconds
plotfluxatend=0;                    % plot the global flux at the end
of program, 1=Yes 0=No

recordPcontour=0;                   % record the contour Pressure
data, 1=Yes 0=No
Pconfrequency=1e-2;                 % record the Pressure contour
every _______ seconds
Pconlevel=-20e6;                    % what pressure level should the
Pressure contour follow? [Pa]

%-------------------------------------------------------------------
------
```

```
%%
% Run the subroutines
S01_TransientFiberMatrix_BuildMesh
S02_TransientFiberMatrix_BuildIndex
S03_TransientFiberMatrix_BuildFiberCluster
S04_TransientFiberMatrix_Run




% S01_TransientFiberMatrix_BuildMesh
% Last Edited 23 June 2015


%-------------------------------------------------------------------
------
% Description
% This code will build a single column of fibers in a subsection of
porous
% domain. This subdomain will have the proper distance along the
axial
% direction of the fibers but it will have a "unit cell width." After
this
% single column of fibers has been generated, a larger mesh will be
created
% to build a mesh which incorporates the total number of desired
columns of
% fibers into the domain. This larger mesh will retain the values of
beta,
% lambda, and alpha from the original "unit column of fibers"

% Revision History
% ----------------
% v01 - created


%-------------------------------------------------------------------
------
% Calculations to find ndx, ndy

% First, we need to find the width of the voids, and find an
appropriate
% value for ndx that will exactly match the void location in the
domain.
% Let n = number of ndx's across ew
% Let m = number of ndx's across vw
vw=beta;                    %distance across the void
ew=(1-beta)/2;              %distance from x'=0 to void on either
side
searchflag=0;
if vw < ew
    %The void width is the smallest dimension
```

```
    m=minnodes;    %start m at minnodes, and increase m, until n is
an integer
    while searchflag==0
        if abs((ew*m/vw)-round((ew*m/vw))) < 1e-9
            n=round(ew*m/vw);
            searchflag=1;
        else
            m=m+1;
        end
    end
else
    %The edge is the smallest dimension
    n=minnodes;    %start n at minnodes, and increase m, until n is
an integer
    while searchflag==0
        if abs((vw*n/ew)-round((vw*n/ew))) < 1e-9
            m=round((vw*n/ew));
            searchflag=1;
        else
            n=n+1;
        end
    end
end
ndx=vw/m;

% Now we need to use the same basic procedure to find a common factor
for
% ndy such that all dimensions can be represented by integers of ndy
% Let a = number of ndy's across vh (void height)
% Let b = number of ndy's across gh (gap height)
vh=alpha/(numv*(1+lambda));
gh=alpha/(numv*(1+(1/lambda)));
gh=gh/2;
searchflag=0;
if gh < vh
    %The gap is smaller than the void height
    b=minnodes;    %start b at minnodes, and increase b, until a is
an integer
    while searchflag==0
        if abs((vh*b/gh)-round((vh*b/gh))) < 1e-9
            a=round((vh*b/gh));
            searchflag=1;
        else
            b=b+1;
        end
    end
else
    %The void is smaller than the gap distance
    a=minnodes;    %start a at minnodes and increase a until b is an
integer
    while searchflag==0
        if abs((a*gh/vh)-round((a*gh/vh))) < 1e-9
            b=round((a*gh/vh));
            searchflag=1;
        else
```

```
            a=a+1;
        end
    end
end
ndy=gh/b;
actual_beta=m/(2*n+m);
actual_lambda=numv*2*b/(numv*a);

%-------------------------------------------------------------------
------
% Creating a mesh for a single column of voids

% P matrix
P=zeros(numv*a+numv*2*b+1,1+2*n+m);

% Build a quick image of the domain
if numv==1
    P(1+b:1+b+a,1+n:1+n+m)=1;
else
    P(1+b:1+b+a,1+n:1+n+m)=1;
    frow=(1+b+a)+(2*b);
    for i=2:numv
        P(frow:frow+a,1+n:1+n+m)=i;
        frow=frow+a+(2*b);
    end
end

%-------------------------------------------------------------------
------
% Build additional columns onto the "P matrix"

if numvx > 1
    Poriginal=P;
    Pcombined=P;
    N=max(max(P));
    for k=1:(numvx-1)
        Pnew=Poriginal;
        for i=1:size(Pnew,1)
            for j=1:size(Pnew,2)
                if Pnew(i,j)~=0
                    Pnew(i,j)=Pnew(i,j)+(N*k);
                end
            end
        end
        COL=size(Pcombined,2);
        Pc2=zeros(size(Pnew,1),COL+size(Pnew,2)-1);
        Pc2(:,1:COL)=Pcombined;
        Pc2(:,COL+1:COL+1+size(Pnew,2)-2)=Pnew(:,2:end);
        Pcombined=Pc2;
    end
    P=Pcombined;
    clear Pcombined Pc2 Poriginal N COL
end
```

```
%-------------------------Changing fibers-----------------------------
----
Nvoids=numv*numvx;
n=1;
k=1;
list=zeros(Nvoids,2);
for i=1:Nvoids
    list(1:Nvoids,1)=[1:1:Nvoids];
    list(i,2)=n+(k-1)*numv;
    k=k+1;
    if k==numvx+1
        k=1;
        n=n+1;
    end
end
for i=1:size(P,1)
    for j=1:size(P,2)
        for x=1:Nvoids
            if P(i,j)==x
                P(i,j)=list(find(list(:,2)==x),1);
                break
            end
        end
    end
end

if showpatternfigure==1
    imagesc(P)
end

%-------------------------------------------------------------------
------
% calculate some actual dimensions in the problem
dx=ndx*W;
dy=ndy*W;

fiber_width=(m-1)*dx;
fiber_height=(a-1)*dy;
fiber_area=fiber_width*fiber_height;
fiber_volume=fiber_area*Hfiber;

volume_of_water_in_pores=fiber_area*porous_depth*porosity;

domain_width=dx*(size(P,2)-1);
domain_height=dy*(size(P,1)-1);




% S02_TransientFiberMatrix_BuildIndex

%Description:
```

```
%This file will determine the kind of nodes, and then add the nodes
%around the fiber to the vp list and the boundaries nodes to the
vbound
%list

Nvoids=numv*numvx; %number of void in the domain

% Build the v and vp and vbound matrices
v=zeros(1,Nvoids);
vp=v;
vpbound=vp;
startrow=v;
endrow=v;
startcol=v;
endcol=v;
for x=1:Nvoids
    vrow=1;
    vprow=1;
    vboundrow=1;
    for i=1:size(P,1)
        for j=1:size(P,2)
            if P(i,j)==x %this location is a void

                %record this index in the v matrix
                v(vrow,P(i,j))=i+(j-1)*size(P,1);
                vrow=vrow+1;

                %Determine what kind of node this is, and then add
the boundary
                %nodes to the vp list
                if P(i-1,j)==0 && P(i,j-1)==0 && P(i+1,j)==P(i,j) &&
P(i,j+1)==P(i,j) %top left
                    vp(vprow,P(i,j))=(i-1)+(j-2)*size(P,1); %top left
corner location
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    startrow(1,P(i,j))=i;
                    startcol(1,P(i,j))=j;
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==0 && P(i,j-1)==P(i,j) &&
P(i+1,j)==P(i,j) && P(i,j+1)==P(i,j) %top
                    vp(vprow,P(i,j))=(i-1)+(j-1)*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==0 && P(i,j-1)==P(i,j) &&
P(i+1,j)==P(i,j) && P(i,j+1)==0 %top right
                    vp(vprow,P(i,j))=(i-1)+j*size(P,1);
                    vpbound(vboundrow,P(i,j))=(i)+(j-1)*size(P,1);
                    endcol(1,P(i,j))=j;
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
```

```
                if P(i-1,j)==P(i,j) && P(i,j-1)==P(i,j) &&
P(i+1,j)==P(i,j) && P(i,j+1)==0 %right
                    vp(vprow,P(i,j))=i+j*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==P(i,j) && P(i,j-1)==P(i,j) &&
P(i+1,j)==0 && P(i,j+1)==0 %bottom right
                    vp(vprow,P(i,j))=(i+1)+j*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==P(i,j) && P(i,j-1)==P(i,j) &&
P(i+1,j)==0 && P(i,j+1)==P(i,j) %bottom
                    vp(vprow,P(i,j))=(i+1)+(j-1)*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==P(i,j) && P(i,j-1)==0 && P(i+1,j)==0 &&
P(i,j+1)==P(i,j) %bottom left
                    vp(vprow,P(i,j))=(i+1)+(j-2)*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    endrow(1,P(i,j))=i;
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
                if P(i-1,j)==P(i,j) && P(i,j-1)==0 &&
P(i+1,j)==P(i,j) && P(i,j+1)==P(i,j) %left
                    vp(vprow,P(i,j))=i+(j-2)*size(P,1);
                    vpbound(vboundrow,P(i,j))=i+(j-1)*size(P,1);
                    vprow=vprow+1;
                    vboundrow=vboundrow+1;
                end
            end
        end
    end
end




% S03_TransientFiberMatrix_BuildFiberCluster
% Last Edited 23 June 2015


%-------------------------------------------------------------------
------
% Description
% The result of A02 is a mesh containing zeros on the porous
substrate or a
```

```
% nonzero number corresponding the identify of the fiber that node is
a
% part of. This code will scan this mesh matrix, P, and build the
Fiber
% cluster, F


% F
% cell array to store information about each fiber
% each row will correspond to one fiber within the domain
% column definitions
% 01 -  index list of every node associated with the fiber
% 02 -  index list of every node associated with the perimeter of the
fiber
% 03 -  index list of every node associated with the porous substrate
%       counterpart to the nodes in column 02. Columns 02 and 03 will
be used to
%       quickly solve for the delta(P) around each fiber
% 04 -  surface tension of the fiber
% 05 -  cavitation flag (0=full / 1=empyting / -1=empyt)
% 06 -  instantaneous volume of water in the fiber
% 07 -  pressure of the fiber
% 08 -  time of cavitation of the fiber
% 09 -  amount of time to empty the fiber
% 10 -  pressure of the fiber just before cavitation
% 11 -  start row
% 12 -  start column
% 13 -  end row
% 14 -  end column


% Revision History
% ----------------
% v01 - created


%-------------------------------------------------------------------
------

%first 3 array of the cell
for i=1:Nvoids
    F{i,1}=v(:,i);
    F{i,2}=vpbound(:,i);
    F{i,3}=vp(:,i);

    sigma_ok=0;
    while sigma_ok < 1
       sigma = mean_sigma + stdev_sigma.*randn(1);
       if sigma > 0;
           sigma_ok=2;
       end
    end
    F{i,4}=sigma;
```

```
    F{i,5}=0;
    F{i,6}=fiber_volume;
    F{i,7}=P_initial;
    F{i,8}=0;
    F{i,9}=0;
    F{i,10}=0;
    F{i,11}=startrow(1,i);
    F{i,12}=startcol(1,i);
    F{i,13}=endrow(1,i);
    F{i,14}=endcol(1,i);
end




% S04_TransientFiberMatrix_Run
% Last Edited 23 June 2015


%-------------------------------------------------------------------------
------


% Revision History
% ----------------
% v01 - created

%-------------------------------------------------------------------------
------

% CONSTANTS
Boltz = 1.3806488e-23;
Planck = 6.62606957e-34;
Temp = 293.15;
Bw=2.2e9;                   %Bulk modulus of water [Pa]

P(:,:)=P_initial;
P(1,:)=P_top;
P(end,:)=P_bottom;
P(:,1)=P(:,2);
P(:,end)=P(:,end-1);
Pnew=P;

% calculate the largest ndt value
C=(kappa*Bw)/(porosity);
dt=(1/(2*C))*(((1/(dx^2))+(1/(dy^2)))^-1);
Cnew=(kappa_new*Bw)/(porosity);
dt_new=(1/(2*Cnew))*(((1/(dx^2))+(1/(dy^2)))^-1);
dt=min(dt,dt_new);
dt=dt/10^3;

% create some common coefficients for the solution
sx=C*dt/(dx^2);
```

```
sy=C*dt/(dy^2);
sx_new=Cnew*dt/(dx^2);
sy_new=Cnew*dt/(dy^2);


cav_count=0;
flags=zeros(size(F,1),1);
diff=1e10;
stop_sim=0;

coeff_dVol = -1*kappa*porous_depth*dt;
coeff_prob1 = (3*Boltz*Temp*dt)/(32*pi()*Planck);
coeff_prob2 = (-16*pi())/(3*Boltz*Temp);


savefileframe=1;
filenum=1;
if showscreenoutputs==1
    showoutputsframe=0;
end

if showpressurefigure==1
    showpressureframe=0;
end

if savepressurefield==1
    savepressureframe=1;
    M{1,1}=0;
    M{1,2}=P;
    MovieRow=2;
end

if recordglobalflux==1
    qin=[0,0];
    qout=[0,0];
    fluxframe=1;
end

if recordPcontour==1
    savePconframe=1;
    Pcon{1,1}=0;
    Pcon{1,2}=contourc(P,[Pconlevel Pconlevel]);
end
%%
t=0;
while stop_sim==0
    t=t+dt;

    %update the Pressure field
    Pnew(2:end-1,2:end-1)=P(2:end-1,2:end-1)+(sx*(P(2:end-1,1:end-2)-
2*P(2:end-1,2:end-1)+P(2:end-1,3:end)))+(sy*(P(1:end-2,2:end-1)-
2*P(2:end-1,2:end-1)+P(3:end,2:end-1)));
```

```
    Pnew(end,:)=P_bottom;                          %top boundary
condition
    Pnew(1,:)=P_top;                               %bottom boundary
condition
    Pnew(2:end-1,1)=Pnew(2:end-1,2);               %left boundary
condition
    Pnew(2:end-1,end)=Pnew(2:end-1,end-1);         %right boundary
condition


    %Now loop through the clusters, update the cluster pressure,
check for
    %cavitation, and update any emptying that may be going on. If the
    %cluster is empty it remains a short-circuit in the pressure.
    for k=1:size(F,1)
        if F{k,5}~=-1 || emptyclusters_are_shortcircuits==1
            v1=F{k,2};
            v2=F{k,3};
            v3=F{k,1};

            sum_dP=0;
            sumdP=zeros(size(v1,1),1);
            sumdP(:,1)=P(v1)-P(v2);
            sum_dP=sum(sumdP);

            dVol = coeff_dVol*sum_dP; %This is total volume leaving
the cluster
        end

        if F{k,5}==0        %cluster has not yet cavitated
            F{k,6}=F{k,6} + dVol;
            total_volume = F{k,6} + volume_of_water_in_pores;
            dP = Bw*dVol/total_volume; %This is the change in
pressure of the cluster
            F{k,7}=F{k,7} + dP; %This is the new pressure inside the
cluster
            %randomly decide if cluster has cavitated
            cavitate=0;

            % CAVITATION DECISION CODE GOES HERE
            Prob=1-
exp(coeff_prob1*F{k,6}*((F{k,7})^3)/((F{k,4})^3)*exp(coeff_prob2*((F{
k,4})^3)/((F{k,7})^2)));
            if rand()<Prob
                cavitate=1;
            end

            if cavitate == 1 %Yes you have cavitated
                F{k,5}=1;               %change flag to emptying
                flags(k,1)=1;

                %///
%                 F{k,5}=-1;              %change flag to emptying
////// ENTRY TO FAKE VOLUME
```

```
%                   cav_count=cav_count+1;  %also added here for FAKE
//////
%                   flags(k,1)=-1;          %also added here for FAKE
//////
                  %///

                  F{k,10}=F{k,7};         %record the pressure at
cavitation
                  F{k,8}=t;               %record the time of
cavitation
                  F{k,7}=0;               %set cluster pressure to zero
              end
              Pnew(v3)=F{k,7}(1,1);     %set the pressure to the same
value everywhere in the cluster
          elseif F{k,5}==1    %cluster has cavitated but is not yet
empty
              F{k,6}=F{k,6} + dVol;
              if F{k,6}<=0 %YAY, the cluster is empty
                  F{k,5}=-1;
                  flags(k,1)=-1;
                  cav_count=cav_count+1;
              else %no, you are still emptying this cluster
                  F{k,9}=F{k,9}+dt;
              end
              Pnew(v3)=F{k,7}(1,1);     %set the pressure to the same
value everywhere in the cluster
          elseif F{k,5}==-1   %cluster is empty
              if emptyclusters_are_shortcircuits==1
                  total_volume = volume_of_water_in_pores;
                  dP = Bw*dVol/total_volume;  %This is the change in
pressure of the cluster
                  F{k,7}=F{k,7} + dP;         %This is the new pressure
inside the cluster
                  Pnew(v3)=F{k,7}(1,1);       %set the pressure to the
same value everywhere in the cluster
              elseif emptyclusters_are_shortcircuits==0
                  %update pressure inside the cluster
                  startrow=F{k,11};
                  startcol=F{k,12};
                  endrow=F{k,13};
                  endcol=F{k,14};
                  %Pressure inside the boundary
                  Pnew(startrow+1:endrow-1,startcol+1:endcol-
1)=P(startrow+1:endrow-1,startcol+1:endcol-
1)+(sx_new*(P(startrow+1:endrow-1,startcol:endcol-2)-
2*P(startrow+1:endrow-1,startcol+1:endcol-1)+P(startrow+1:endrow-
1,startcol+2:endcol)))+(sy_new*(P(startrow:endrow-
2,startcol+1:endcol-1)-2*P(startrow+1:endrow-1,startcol+1:endcol-
1)+P(startrow+2:endrow,startcol+1:endcol-1)));
                  %Pressure around the boundary
                  Pnew(startrow+1:endrow-
1,startcol)=(1/(kappa_new+kappa))*(kappa_new*(P(startrow+1:endrow-
1,startcol+1))+(kappa*P(startrow+1:endrow-1,startcol-1))); %left
```

```
                Pnew(startrow,startcol+1:endcol-
1)=(1/(kappa_new+kappa))*(kappa_new*(P(startrow+1,startcol+1:endcol-
1))+(kappa*P(startrow-1,startcol+1:endcol-1))); %top
                Pnew(startrow+1:endrow-
1,endcol)=(1/(kappa_new+kappa))*(kappa*(P(startrow+1:endrow-
1,endcol+1))+(kappa_new*P(startrow+1:endrow-1,endcol-1))); %right
                Pnew(endrow,startcol+1:endcol-
1)=(1/(kappa_new+kappa))*(kappa*(P(endrow+1,startcol+1:endcol-
1))+(kappa_new*P(endrow-1,startcol+1:endcol-1))); %bottom
            end
        end
    end

    %Set the old pressure equal to the new pressure
    P(:,:)=Pnew(:,:);

    %Check to see when simulation should end
    if finalruntime>0
        if t>finalruntime
            stop_sim=1;
        end
    else
        if max(flags)==-1
            stop_sim=1;
        end
        if diff < 0.5
            stop_sim=1;
        end
    end

    %Calculate and record the flux if needed
    if recordglobalflux==1
        if t>=fluxfrequency*fluxframe
            fluxin=zeros(size(P,2),1);
            fluxin=sum(P(end,:)-P(end-1,:))/size(P,2)/dy;
            k=size(qin,1)+1;
            qin(k,1:2)=[t,kappa*fluxin];

            fluxout=zeros(size(P,2),1);
            fluxout=sum(P(2,:)-P(1,:))/size(P,2)/dy;
            k=size(qout,1)+1;
            qout(k,1:2)=[t,kappa*fluxout];

            diff=abs(100*(fluxin-fluxout)/fluxin);

            fluxframe=fluxframe+1;
        end
    end

    %Calculate and record the Pressure contour if needed
    if recordPcontour==1
        if t>=Pconfrequency*savePconframe
            savePconframe=savePconframe+1;
            k=size(Pcon,1)+1;
```

```
            Pcon{k,1}=t;
            Pcon{k,2}=contourc(P,[Pconlevel Pconlevel]);
        end
    end

    %saving and outputs
    if showscreenoutputs==1
        if t>showoutputsfrequency*showoutputsframe
            clc
            t
            cav_fraction=cav_count/size(F,1)
            diff

            showoutputsframe=showoutputsframe+1;
        end
    end
    if showpressurefigure==1
        if t>showpressureframe*showpressurefrequency
            if showpressureDtype==2
                figure(2)
                clim = [P_top 0];
                imagesc(P,clim), colorbar('location','eastoutside')
                if recordPcontour==1
                    hold on
                    contour(P,[Pconlevel Pconlevel], '--w',
'LineWidth', 3)
                    hold off
                end
            elseif showpressureDtype==3
                figure(2)
                surf(P)
                drawnow
            end
            showpressureframe=showpressureframe+1;
        end
    end
    if savepressurefield==1
        if t>savepressurefrequency*savepressureframe
            M{MovieRow,1}=t;
            M{MovieRow,2}=P;
            MovieRow=MovieRow+1;
            savepressureframe=savepressureframe+1;
        end
    end
    if t>savefilefrequency*savefileframe
        fname=sprintf('%s_%i',filename,filenum);
        save(fname)
        clear M
        filenum=filenum+1;
        savefileframe=savefileframe+1;
    end
end

if plotfluxatend==1
    %%
```

```
    figure()
    plot(qin(:,1),qin(:,2),'-b')
    hold on
    plot(qout(:,1),qout(:,2),'-r')
    xlabel('time [s]')
    ylabel('fluid flux [m s^-^1]')
    legend('flux in','flux out')
end
%%
fname=sprintf('%s_%s',filename,'end');
save(fname)
```

Appendix 12

XII. Matlab code to solve for cavitation dynamics in the effective medium model

```
% Effective Medium Equations

clear all; close all; clc

%INPUTS
neta=0.5/2;                             %area ratio = (beta)/(1 +
old_lambda)
well_density=1;                         %total well density at every node
z=5e-6;                                 %depth of poSi [m]
poro=0.5;                               %porosity of poSi [-]
B=2.2e9;                                %bulk modulos of liquid [Pa]
Ks=1.9e-17;                             %permeability of the substrate
[m^2 Pa^-1 s^-1]
d=2;                                    %number of dimensions in Maxwell
permeability [-]
Vw=(100e-6)*(100e-6)*(20e-8);           %volume of a single well [m3]
Aw=(100e-6)*(100e-6);                   %area of a single well [m2]
domain_aspect_ratio=1;                  %domain length in y-
direction/domain length in x-direction
sigma=0.019;                            %surface tension of water [N m^-
1]
k_boltz=1.3806488e-23;                  %Boltzmann constant [m2 kg s^-2
K^-1]
Temp=293;                               %temperature [K]
Psat=0;                                 %saturation pressure [Pa]
h_planck=6.62606957e-34;                %Planck's constant [m2 kg s^-1]

P_right=-45e6;
P_left=0;

record_freq=0.1;                          %Add data to the M cell every
_____ seconds

%%
% Geometry calculations

A=Aw*well_density/neta;                   %domain area over which you
are averaging [m^2]
dx=sqrt(A);                               %distance between nodes in
the x direction [m]
dy=dx;                                    %distance between nodes in
the y direction [m]
length_x=11*dx;                           %length of the domain in the
x direction [m]
length_y=domain_aspect_ratio*length_x;    %length of the domain in the
y direction [m]
Rw=(Aw/pi())^(1/2);                       %well radius
```

```
%%
%Variables
gama=8;
dt=((dx^2)*poro)/((2^gama)*B*Ks);
t=0;
nodesx=round(((length_x)/dx)+1);
nodesy=round(((length_y)/dy)+1);
P=zeros(nodesx,nodesy);
Nf=P;
Ne=P;
Na=P;
Nt=P;
%Ke=P;

%Initial Conditions (t=0)
Nf(:,:)=well_density/A;
Nt=Nf;

%Boundaries conditions [Pa]
P(1,:)=P(end,:);
P(:,end)=(2*P_right)-P(:,end-1);
P(:,1)=(2*P_left)-P(:,2);

Pnew=P;
Nfnew=Nf;
Nanew=Na;
Nenew=Ne;
stop_flag=0;
diff=1e10;

%Calculating Lbound
count=dy;
av_length=round(nodesy/2);
count1=dy*(nodesy-av_length);
L_bound=P(2:end-1,2:end-1);
for frame=1:nodesy-2
    if frame<=av_length-2
        L_bound(:,frame)=count;
        count=count+dy;
    else
        L_bound(:,frame)=count1;
        count1=count1-dy;
    end
end

recordframe=1;
M{recordframe,1}=('time');
M{recordframe,2}=('P field');
M{recordframe,3}=('Nf');
M{recordframe,4}=('Na');
M{recordframe,5}=('Ne');
M{recordframe,6}=('Keff');
recordframe=2;
```

```
M{recordframe,1}=t;
M{recordframe,2}=P;
M{recordframe,3}=Nf;
M{recordframe,4}=Na;
M{recordframe,5}=Ne;
M{recordframe,6}=0;

recordframe=1;


%%
%Equations
while stop_flag==0
    %%
    clc
    t=t+dt
    diff

    %Average of nf_x and na_x
    Nfxin=(Nf(2:end-1,2:end-1)+Nf(2:end-1,1:end-2))/2;
    Naxin=(Na(2:end-1,2:end-1)+Na(2:end-1,1:end-2))/2;
    Nfxout=(Nf(2:end-1,2:end-1)+Nf(2:end-1,3:end))/2;
    Naxout=(Na(2:end-1,2:end-1)+Na(2:end-1,3:end))/2;
    Ntxin=(Nt(2:end-1,2:end-1)+Nt(2:end-1,1:end-2))/2;
    Ntxout=(Nt(2:end-1,2:end-1)+Nt(2:end-1,3:end))/2;

    %Ke_x calculation

Kex_in=Ks*(1+((d*((((Nfxin(:,:)+Naxin(:,:))./(Ntxin(:,:)))*neta))./(
1-(((Nfxin(:,:)+Naxin(:,:))./Ntxin(:,:))*neta))));

Kex_out=Ks*(1+((d*((((Nfxout(:,:)+Naxout(:,:))./(Ntxout(:,:)))*neta))
)./(1-(((Nfxout(:,:)+Naxout(:,:))./Ntxout(:,:))*neta))));

    %Average of nf_y and na_y
    Nfyin=(Nf(2:end-1,2:end-1)+Nf(1:end-2,2:end-1))/2;
    Nayin=(Na(2:end-1,2:end-1)+Na(1:end-2,2:end-1))/2;
    Nfyout=(Nf(2:end-1,2:end-1)+Nf(3:end,2:end-1))/2;
    Nayout=(Na(2:end-1,2:end-1)+Na(3:end,2:end-1))/2;
    Ntyin=(Nt(2:end-1,2:end-1)+Nt(1:end-2,2:end-1))/2;
    Ntyout=(Nt(2:end-1,2:end-1)+Nt(3:end,2:end-1))/2;

    %Ke_y calculation

Key_in=Ks*(1+((d*((((Nfyin(:,:)+Nayin(:,:))./(Ntyin(:,:)))*neta))./(
1-(((Nfyin(:,:)+Nayin(:,:))./Ntyin(:,:))*neta))));

Key_out=Ks*(1+((d*(((((Nfyout(:,:)+Nayout(:,:))./(Ntyout(:,:)))*neta)
))./(1-(((Nfyout(:,:)+Nayout(:,:))./Ntyout(:,:))*neta)))));

    %Flux method
    Jxin=(-1*Kex_in(:,:)/poro).*((P(2:end-1,2:end-1)-P(2:end-1,1:end-
2))/dx);
```

```
    Jxout=(-1*Kex_out(:,:)/poro).*((P(2:end-1,3:end)-P(2:end-1,2:end-
1))/dx);
    Jyin=(-1*Key_in(:,:)/poro).*((P(2:end-1,2:end-1)-P(1:end-2,2:end-
1))/dy);
    Jyout=(-1*Key_out(:,:)/poro).*((P(3:end,2:end-1)-P(2:end-1,2:end-
1))/dy);

    %Pressure calculation
    Ke(:,:)=Ks*(1+((d*(((Nf(2:end-1,2:end-1)+Na(2:end-1,2:end-
1))./(Nt(2:end-1,2:end-1)))*neta)./((1-((Nf(2:end-1,2:end-
1)+Na(2:end-1,2:end-1))./(Nt(2:end-1,2:end-1)))*neta))));
    C1=((1/B)+((Na(2:end-1,2:end-1)+Nf(2:end-1,2:end-
1))*Vw)/(poro*z*B));
    Lscreen=1./((Na(2:end-1,2:end-1)).^(1/2));
    Rc=min(Lscreen,L_bound);
    h=real((2*pi()*Ke(:,:)*z)./(log(Rc(:,:)./Rw)));

    Pnew(2:end-1,2:end-1)=P(2:end-1,2:end-
1)+((dt./C1(:,:)).*((((Jxin(:,:)-Jxout(:,:))/dx)+((Jyin(:,:)-
Jyout(:,:))/dy))-(Na(2:end-1,2:end-1).*h(:,:).*P(2:end-1,2:end-
1))/poro*A*z));
    Pnew(1,:)=Pnew(2,:); %no flux top
    Pnew(end,:)=Pnew(end-1,:); %no flux bottom
    %diff=sum(sum(abs(Pnew-P)))/(size(P,1)*size(P,2))/dt;
    P(:,:)=real(Pnew(:,:));

    P(:,end)=(2*P_right)-P(:,end-1);
    P(:,1)=(2*P_left)-P(:,2);

    qout=-1*Kex_out(2,end)*sum(P(:,end)-P(:,end-1))/size(P,1);
    qin=-1*Kex_in(2,1)*sum(P(:,2)-P(:,1))/size(P,1);
    diff=abs(qout-qin)*100/qout;

    %Alpha calculation
    Lo=real((-3*k_boltz*Temp*((P(:,:)-
Psat).^3))./(32*pi()*h_planck*sigma^3));
    L=real(Lo.*(exp((-16*pi()*(sigma^3))./(3*k_boltz*Temp*((P(:,:)-
Psat).^2)))));
    alpha=real(Vw.*L(:,:));

    %Update Na, Ne, Nf
    Nfnew(2:end-1,2:end-1)=Nf(2:end-1,2:end-1).*exp((-alpha(2:end-
1,2:end-1)*dt));
    Nenew(2:end-1,2:end-1)=Ne(2:end-1,2:end-1)+dt.*(-1*P(2:end-
1,2:end-1).*Na(2:end-1,2:end-1).*h(:,:)./Vw);
    Nanew(:,:)=Nt(:,:)-(Nfnew(:,:)+Nenew(:,:));

    %Update the boundary conditions of Na, Ne, Nf
    Na(:,:)=real(Nanew(:,:));
    Nf(:,:)=real(Nfnew(:,:));
    Ne(:,:)=real(Nenew(:,:));

    Nf(:,1)=Nf(:,2);
    Na(:,1)=Na(:,2);
```

```
    Ne(:,1)=Ne(:,2);

    Nf(:,end)=Nf(:,end-1);
    Na(:,end)=Na(:,end-1);
    Ne(:,end)=Ne(:,end-1);

    Nf(1,:)=Nf(2,:);
    Na(1,:)=Na(2,:);
    Ne(1,:)=Ne(2,:);

    Nf(end,:)=Nf(end-1,:);
    Na(end,:)=Na(end-1,:);
    Ne(end,:)=Ne(end-1,:);

    if t > recordframe*record_freq
        j=size(M,1)+1;
        M{j,1}=t;
        M{j,2}=P;
        M{j,3}=Nf;
        M{j,4}=Na;
        M{j,5}=Ne;
        M{j,6}=Ke./Ks;
        M{j,7}=diff;
        recordframe=recordframe+1;
    end
%     if diff < 0.5
%         stop_flag=1;
%     end
    if t >= 100
        stop_flag=1;
    end
end

save effective_medium_45MPa_100sec_NEWdx
```

REFERENCES

Akinnikawe, O., Chaudhary, A., Vasquez, O., Enih, C., Ehlig-Economides, C., 2010. Increasing CO2-Storage Efficiency through a CO2-Brine Displacement Approach. Proc. SPE Int. Conf. CO2 Capture, Storage, Util. doi:10.2118/139467-MS

Alpak, F.O., Lake, L.W., Embid, S.M., Intevep, S.A., 1999. Validation of a Modified Carman-Kozeny Equation To Model Two-Phase Relative Permeabilities 1–11.

Anchliya, A., Ehlig-Economides, C., 2009. Aquifer Management to Accelerate CO 2 Dissolution and Trapping 2–4.

Aspnes, D.E., 1981. Local-field effects and effective-medium theory:A microscopic perspective. Am. J. Phys. 50.

Bachu, S., 2013. Drainage and Imbibition CO2/Brine Relative Permeability Curves at in Situ Conditions for Sandstone Formations in Western Canada. Energy Procedia 37, 4428–4436. doi:10.1016/j.egypro.2013.07.001

Bejan, A., 2006. Convection in Porous Media.

Bennion, D.B., Bachu, S., 2006a. Dependence on Temperature , Pressure , and Salinity of the IFT and Relative Permeability Displacement Characteristics of CO 2 Injected in Deep Saline Aquifers. pp. 1–9.

Bennion, D.B., Bachu, S., 2006b. Supercritical CO 2 and H 2 S — Brine Drainage and Imbibition Relative Permeability Relationships for Intergranular Sandstone and Carbonate Formations, in: SPE 99326.

Bennion, D.B., Energy, H., Bachu, S., Resources, A.E., 2008. Drainage and Imbibition Relative Permeability Relationships for Supercritical CO 2 / Brine and H 2 S /

Brine Systems in Intergranular Sandstone , Carbonate , Shale , and Anhydrite Rocks 487–496.

Berkowitz, B., Ewing, R.P., 1998. PERCOLATION THEORY AND NETWORK MODELING. Surv. Geophys. 19, 23–72.

Brenner, H., 1980. Dispersion Resulting From Flow Through Spatially Periodic Porous Media. Philos. Trans. R. Soc. A 297.

Burnside, N.M., Naylor, M., 2014. Review and implications of relative permeability of CO2/brine systems and residual trapping of CO2. Int. J. Greenh. Gas Control 23, 1–11. doi:10.1016/j.ijggc.2014.01.013

Buscheck, T. a., Sun, Y., Chen, M., Hao, Y., Wolery, T.J., Bourcier, W.L., Court, B., Celia, M. a., Julio Friedmann, S., Aines, R.D., 2012. Active CO2 reservoir management for carbon storage: Analysis of operational strategies to relieve pressure buildup and improve injectivity. Int. J. Greenh. Gas Control 6, 230–245. doi:10.1016/j.ijggc.2011.11.007

Choy, T.C., 1999. Effective Medium Theory: Principles and Applications. Clarendon Press, Oxford.

Corey, A.T., 1954. The Interrealation Between Gas and Oil Relative Permeabilities. Prod. Mon. 19, 38–41.

Darcy, H., 1856. Les Fontaines Publiques de la Ville de Dijon. Victor Dalmont, Paris.

Delshad, M., Wheeler, M., Kong, X., 2010. A Critical Assessment of CO2 Injection Strategies in Saline Aquifers. Proc. SPE West. Reg. Meet. 27–29. doi:10.2118/132442-MS

DOE, 2012. The United States 2012 Carbon Utilization and Storage Atlas, Fourth. ed.

US DOE Office of Fossil Energy.

Dooley, J.J., Dahowski, R., Davidson, C., 2005. The Midwest Regional Carbon Sequestration (MRCSP) Phase I Final Report.

Doughty, C., 2010. Investigation of CO2 Plume Behavior for a Large-Scale Pilot Test of Geologic Carbon Storage in a Saline Formation. Transp. Porous Media 82, 49–76. doi:10.1007/s11242-009-9396-z

Doughty, C., 2007. Modeling geologic storage of carbon dioxide: Comparison of non-hysteretic and hysteretic characteristic curves. Energy Convers. Manag. 48, 1768–1781. doi:10.1016/j.enconman.2007.01.022

Doughty, C., Myer, L.R., Oldenburg, C.M., 2009. Predictions of long-term behavior of a large-volume pilot test for CO2 geological storage in a saline formation in the Central Valley, California. Energy Procedia 1, 3291–3298. doi:10.1016/j.egypro.2009.02.115

Ennis-King, J., Paterson, L., 2007. Coupling of geochemical reactions and convective mixing in the long-term geological storage of carbon dioxide. Int. J. Greenh. Gas Control 1, 86–93. doi:10.1016/S1750-5836(07)00034-5

Erendi, A., Cathles, L.M., 2001. Gas Capillary Inhibition to Oil Production 607–618.

Ferer, M., Bromhal, G.S., Smith, D.H., 2003. Pore-level modeling of immiscible drainage: validation in the invasion percolation and DLA limits. Phys. A Stat. Mech. its Appl. 319, 11–35. doi:10.1016/S0378-4371(02)01508-X

Fey, T., 2015. Glass Ceramics [WWW Document]. URL http://www.glass-ceramics.uni-erlangen.de/Research/Simulation/project_sim1.htmand Storage

Figueroa, J.D., Fout, T., Plasynski, S., McIlvried, H., Srivastava, R.D., 2008. Advances

in CO2 capture technology-The U.S. Department of Energy's Carbon Sequestration Program. Int. J. Greenh. Gas Control 2, 9–20. doi:10.1016/S1750-5836(07)00094-1

Finley, R., 2005. An Assessment of Geological Carbon Sequestration Options in the Illinois Basin.

Fokas, A.S., Yortsos, Y.C., 1982. On the Exactly Solvable Equation Occurring in Two-Phase Flow in Porous Media. J. Appl. Math. 42, 318–332.

Freeze, R.A., 1994. Henry Darcy and the fountains of Dijon. Ground Water. doi:10.1111/j.1745-6584.1994.tb00606.x

Gilfillan, S.M. V, Lollar, B.S., Holland, G., Blagburn, D., Stevens, S., Schoell, M., Cassidy, M., Ding, Z., Zhou, Z., Lacrampe-Couloume, G., Ballentine, C.J., 2009. Solubility trapping in formation water as dominant CO(2) sink in natural gas fields. Nature 458, 614–8. doi:10.1038/nature07852

Golding, M.J., Neufeld, J. a., Hesse, M. a., Huppert, H.E., 2011. Two-phase gravity currents in porous media. J. Fluid Mech. 678, 248–270. doi:10.1017/jfm.2011.110

Grainger, C.A., Kolstad, C.D., 2010. Who pays a price on carbon? Environ. Resour. Econ. 46, 359–376. doi:10.1007/s10640-010-9345-x

Haberman, R., 2004. Applied Partial Differential Equations with Fourier Series and Boundary Value Problems, 4th ed. Prentice Hall.

Hesse, M. a., Orr, F.M., Tchelepi, H. a., 2008. Gravity currents with residual trapping. J. Fluid Mech. 611, 35–60. doi:10.1017/S002211200800219X

Hesse, M. a., Tchelepi, H. a., Cantwel, B.J., Orr, F.M., 2007. Gravity currents in horizontal porous layers: transition from early to late self-similarity. J. Fluid Mech.

577, 363. doi:10.1017/S0022112007004685

Huber, E.J., Stroock, A.D., Koch, D.L., 2016. Analysis of a time dependent injection strategy to accelerate the residual trapping of sequestered CO2 in the geologic subsurface. Int. J. Greenh. Gas Control 44, 185–198. doi:10.1016/j.ijggc.2015.11.024

Hunt, A.G., 2005. Percolation Theory for Flow in Porous Media. Springer, Berlin.

Hunt, A.G., Ewing, R.P., Ghanbarian, B., 2014. Percolation Theory for Flow in Porous Media, Third. ed. Springer, London.

Huppert, H.E., Neufeld, J. a., 2014. The Fluid Mechanics of Carbon Dioxide Sequestration. Annu. Rev. Fluid Mech. 46, 255–272. doi:10.1146/annurev-fluid-011212-140627

Jordan, P., Doughty, C., 2009. Sensitivity of CO2 migration estimation on reservoir temperature and pressure uncertainty. Energy Procedia 1, 2587–2594. doi:10.1016/j.egypro.2009.02.024

Juanes, R., MacMinn, C.W., Szulczewski, M.L., 2010. The footprint of the CO2 plume during carbon dioxide storage in saline aquifers: Storage efficiency for capillary trapping at the basin scale. Transp. Porous Media 82, 19–30. doi:10.1007/s11242-009-9420-3

Juanes, R., Spiteri, E.J., Orr, F.M., Blunt, M.J., 2006. Impact of relative permeability hysteresis on geological CO 2 storage. Water Resour. Res. 42, n/a-n/a. doi:10.1029/2005WR004806

Khaninezhad, M.M., Jafarpour, B., Li, L., 2012. Sparse geologic dictionaries for subsurface flow model calibration: Part II. Robustness to uncertainty. Adv. Water

Resour. 39, 122–136. doi:10.1016/j.advwatres.2011.10.005

Kopp, A., Binning, P.J., Johannsen, K., Helmig, R., Class, H., 2010. A contribution to risk analysis for leakage through abandoned wells in geological CO2 storage. Adv. Water Resour. 33, 867–879. doi:10.1016/j.advwatres.2010.05.001

Lafitte, T., Mendiboure, B., Piñeiro, M.M., Bessières, D., Miqueu, C., 2010. Interfacial properties of water/CO2: a comprehensive description through a Gradient Theory-SAFT-VR Mie approach. J. Phys. Chem. B 114, 11110–6. doi:10.1021/jp103292e

Lage, J.L., 1998. Transport Phenomena in Porous Media. Pergamon, Netherlands.

Land, C., 1968. Calculation of Imbibition Relative Permeability for Two- and Three-Phase Flow From Rock Properties. Soc. Pet. Eng. J. 8. doi:10.2118/1942-PA

Larson, R.G., Scriven, L.E., Davis, H.T., 1981. Percolation Theory of Two Phase Flow in Porous Media. Chem. Eng. Sci. 36, 57–73.

Leonenko, Y., Keith, D.W., 2008. Reservoir engineering to accelerate the dissolution of CO2 stored in aquifers. Environ. Sci. Technol. 42, 2742–7.

Leverett, M.C., 1941. Capillary Behavior in Porous Solids. Trans. AIME 142, 152–169. doi:10.2118/941152-G

Levy, O., Stroud, D., 1997. Maxwell Garnett theory for mixtures of anisotropic inclusions: Application to conducting polymers. Phys. Rev. B 56, 8035–8046. doi:10.1103/PhysRevB.56.8035

Li, L., Jafarpour, B., 2010. A sparse Bayesian framework for conditioning uncertain geologic models to nonlinear flow measurements. Adv. Water Resour. 33, 1024–1042. doi:10.1016/j.advwatres.2010.06.005

Linstrom, P.J., Mallard, W.G. (Eds.), 2011. NIST Chemistry WebBook, NIST Standard

Reference Database Number 69. National Institute of Standards and Technology, Gaithersburg, MD, 20899.

Liu, J. (Daniel), Sue, H.-J., Thompson, Z.J., Bates, F.S., Dettloff, M., Jacob, G., Verghese, N., Pham, H., 2008. Nanocavitation in Self-Assembled Amphiphilic Block Copolymer-Modified Epoxy. Macromolecules 41, 7616–7624. doi:10.1021/ma801037q

Marchesin, D., Plohr, B.J., 2001. Wave Structure in WAG Recovery. SPE J. 209–219.

Maxwell Garnett, J.C., 1904. XII. Colous in Metal Glasses and in Metallic Films. Philos. Trans. R. Soc. A 203. doi:10.1098/rsta.1948.0007

Meyer, L., 2007. Carbon Sequestration Options for the West Coast States.

Moghanloo, R.G., 2012. Applying Method of Characteristics to Model the Flow of Compressible CO 2 in Aquifers, in: SPE Western Regional Meeting. pp. 1–23.

Müller, N., 2010. Supercritical CO2-Brine Relative Permeability Experiments in Reservoir Rocks—Literature Review and Recommendations. Transp. Porous Media 87, 367–383. doi:10.1007/s11242-010-9689-2

NETL, DOE, 2010. 2010 Carbon Sequestration Atlas of the United States and Canada.

Neufeld, J. a., Huppert, H.E., 2009. Modelling carbon dioxide sequestration in layered strata. J. Fluid Mech. 625, 353. doi:10.1017/S0022112008005703

Pegler, S.S., Huppert, H.E., Neufeld, J. a., 2014. Fluid migration between confined aquifers. J. Fluid Mech. 757, 330–353. doi:10.1017/jfm.2014.469

PetraSIM, 2015. PetraSIM.

Rodosta, T., Ackiewicz, M., Albenze, E., 2014. Status Update and Results from the U.S. Department of Energy Regional Carbon Sequestration Partnership Initiative.

Energy Procedia 63, 6039–6052. doi:10.1016/j.egypro.2014.11.637

Rutqvist, J., Tsang, C.-F., 2002. A study of caprock hydromechanical changes associated with CO 2 -injection into a brine formation. Environ. Geol. 42, 296–305. doi:10.1007/s00254-001-0499-2

Shamshiri, H., Jafarpour, B., 2012. Controlled CO 2 injection into heterogeneous geologic formations for improved solubility and residual trapping. Water Resour. Res. 48, n/a-n/a. doi:10.1029/2011WR010455

Shosa, J.D., Cathles, L.M., 2001. Experimental Investigation of Capillary Blockage of Two Phase Flow in Layered Porous Media, in: GCSSEPM Foundation 21st Annual Research Conference.

Shukla, R., Ranjith, P., Haque, A., Choi, X., 2010. A review of studies on CO2 sequestration and caprock integrity. Fuel 89, 2651–2664. doi:10.1016/j.fuel.2010.05.012

Silin, D., Patzek, T., Benson, S.M., 2009. A Model of Buoyancy-Driven Two-Phase Countercurrent Fluid Flow. Transp. Porous Media 76, 449–469. doi:10.1007/s11242-008-9257-1

Spycher, N., Pruess, K., 2009. A Phase-Partitioning Model for CO2–Brine Mixtures at Elevated Temperatures and Pressures: Application to CO2-Enhanced Geothermal Systems. Transp. Porous Media 82, 173–196. doi:10.1007/s11242-009-9425-y

Spycher, N., Pruess, K., Ennis-King, J., 2003. CO2-H2O mixtures in the geological sequestration of CO2. I. Assessment and calculation of mutual solubilities from 12 to 100°C and up to 600 bar. Geochim. Cosmochim. Acta 67, 3015–3031. doi:10.1016/S0016-7037(03)00273-4

Stauffer, D., 1985. Introduction to Percolation Theory. Taylor & Francis, Philadelphia.

Steadman, E.N., Daly, D.J., Silva, L.L. De, Harju, J.A., Jensen, M.D., Peck, W.D., Smith, S.A., Sorensen, J.A., 2006. PLAINS CO2 REDUCTION (PCOR) PARTNERSHIP (PHASE I) FINAL REPORT.

van Genuchten, M.T., 1980. Closed-form Equation for Predicting the Hydraulic Conductivity of Unsaturated Soils. Soil Sci. Soc. Am. J. 8, 892–898.

Vilarrasa, V., Bolster, D., Dentz, M., Olivella, S., Carrera, J., 2010. Effects of CO2 Compressibility on CO2 Storage in Deep Saline Aquifers. Transp. Porous Media 85, 619–639. doi:10.1007/s11242-010-9582-z

Vincent, O., Sessoms, D. a., Huber, E.J., Guioth, J., Stroock, A.D., 2014. Drying by Cavitation and Poroelastic Relaxations in Porous Media with Macroscopic Pores Connected by Nanoscale Throats. Phys. Rev. Lett. 113, 134501. doi:10.1103/PhysRevLett.113.134501

Wickstrom, L.H., Venteris, E.R., Harper, J.A., Mcdonald, J., Slucher, E.R., Carter, K.M., Greb, S.F., Wells, J.G., Harrison III, W.B., Nuttall, B.C., Riley, R.A., Drahovzal, J.A., Rupp, J.A., Avary, K.L., Lanham, S., Barnes, D.A., Gupta, N., Baranoski, M.A., Radhakkrishnan, P., Solis, M.P., Baum, G.R., Powers, D., Hohn, M.E., Parris, M.P., McCoy, K., Grammer, G.M., Pool, S., Luckhardt, C., Kish, P., 2005. Characterization of Geologic Sequestration Opportunities in the MRCSP Region Phase I Task Report Period of Performance : October 2003 – September 2005.

Wilkinson, D., Willemsen, J.F., 1983. Invasion percolation: a new form of percolation theory. J. Phys. A. Math. Gen. 16, 3365–3376. doi:10.1088/0305-4470/16/14/028

Wollenweber, J., Alles, S., Busch, a., Krooss, B.M., Stanjek, H., Littke, R., 2010. Experimental investigation of the CO2 sealing efficiency of caprocks. Int. J. Greenh. Gas Control 4, 231–241. doi:10.1016/j.ijggc.2010.01.003

Xu, T., Kharaka, Y.K., Doughty, C., Freifeld, B.M., Daley, T.M., 2010. Reactive transport modeling to study changes in water chemistry induced by CO2 injection at the Frio-I Brine Pilot. Chem. Geol. 271, 153–164. doi:10.1016/j.chemgeo.2010.01.006

Yang, H., Xu, Z., Fan, M., Slimane, R.B., Bland, A.E., Wright, I., 2008. Progress in carbon dioxide seperation and capture: A review. J. Environ. Sci. 20, 14–27.

Zimmerman, R.W., 1996. Effective conductivity of a two-dimensional medium containing elliptical inhomogeneities. Proc. R. Soc. Lond. A 452, 1713–1727.

Zunsheng, J., Lifa, Z., Runmin, G., Tingting, L., Hong, W., Wang, H., McLaughlin, F., Bentley, R., Quillinan, S., 2014. Opportunity and Challenges of Integrated Enhanced Oil Recovery Using CO2 Flooding with Geological CO2 Storage in the Ordos Basin, China. Energy Procedia 63, 7761–7771. doi:10.1016/j.egypro.2014.11.810